

PATRÍCIA PRADO BELFIORE

**SCATTER SEARCH PARA PROBLEMAS DE ROTERIZAÇÃO DE
VEÍCULOS COM FROTA HETEROGENEA, JANELAS DE TEMPO E
ENTREGAS FRACIONADAS**

Tese apresentada à Escola Politécnica da
Universidade de São Paulo para a obtenção do título
de Doutor em Engenharia.

São Paulo

2006

PATRÍCIA PRADO BELFIORE

**SCATTER SEARCH PARA PROBLEMAS DE ROTERIZAÇÃO DE
VEÍCULOS COM FROTA HETEROGRÊNEA, JANELAS DE TEMPO E
ENTREGAS FRACIONADAS**

Tese apresentada à Escola Politécnica da Universidade de São Paulo para a obtenção do título de Doutor em Engenharia.

Área de concentração:
Engenharia de Produção

Orientador: Prof. Dr. Hugo T. Y. Yoshizaki

São Paulo

2006

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

São Paulo, 30 de março de 2006.

Assinatura do autor

Assinatura do orientador

FICHA CATALOGRÁFICA

Belfiore, Patrícia Prado

Scatter search para Problemas de Roteirização de Veículos
com Frota Heterogênea, Janelas de Tempo e Entregas
Fracionadas. São Paulo, 2006.

203 p.

Tese de Doutorado – Escola Politécnica da Universidade de
São Paulo. Departamento de Engenharia de Produção.

1. Roteirização 2. Pesquisa Operacional 3. Heurística
4. Varejo I. Universidade de São Paulo. Escola Politécnica.
Departamento de Engenharia de Produção II. t.

Dedico,

*À minha mãe, ao meu pai (em
memória) e ao meu irmão que sempre me
incentivaram.*

*Ao meu noivo Luiz Paulo por todo
apoio, amor e carinho.*

AGRADECIMENTOS

Ao Prof. Hugo T. Y. Yoshizaki pela orientação e contribuição no desenvolvimento deste trabalho.

À toda minha família pelo apoio e incentivo.

À Profa. Débora Pretti Ronconi pelas valiosas dicas que auxiliaram o meu ingresso no mestrado e doutorado, além do aprendizado em sua disciplina que contribuiu para o avanço deste trabalho.

Ao Prof. Vinícius Amaral Armentano e ao Prof. Cláudio Barbieri da Cunha pela contribuição no trabalho.

Aos amigos e professores da Escola Politécnica e de outras unidades da USP pela oportunidade de aprendizado.

À Capes pelo suporte recebido.

*O caminho real da descoberta não
está na procura de passagens
novas, mas em possuir novos
olhos...*

Marcel Proust

SUMÁRIO

LISTA DE FIGURAS	iv
LISTA DE QUADROS	v
LISTA DE TABELAS	vi
LISTA DE TABELAS - ANEXOS	vii
LISTA DE GRÁFICOS	ix
RESUMO	x
ABSTRACT	xii
INTRODUÇÃO	1
1 – A IMPORTÂNCIA DOS PROBLEMAS DE ROTEIRIZAÇÃO DE VEÍCULOS	1
2 – OBJETIVO	3
3 – JUSTIFICATIVA DO TEMA E CONTRIBUIÇÕES	4
4 – DELINEAMENTO DO TRABALHO	5
CAPÍTULO 1 - TAXONOMIA DOS PROBLEMAS DE ROTEIRIZAÇÃO DE VEÍCULOS	7
1.1 - O PROBLEMA BÁSICO DE ROTEIRIZAÇÃO DE VEÍCULOS E SUAS EXTENSÕES	7
1.2 - CLASSIFICAÇÃO DOS PROBLEMAS DE ROTEIRIZAÇÃO DE VEÍCULOS	11
1.2.1 - Classificação segundo Bodin et al. (1983)	
1.2.2 - Classificação segundo Ronen (1988)	
1.2.3 - Classificação segundo Assad (1988)	
1.2.4 - Classificação segundo Desrochers, Lenstra e Savelsbergh (1990)	
1.3 - PROBLEMAS CLÁSSICOS DE ROTEIRIZAÇÃO DE VEÍCULOS	18
1.4 - SÍNTESE DOS PROBLEMAS DE ROTEIRIZAÇÃO DE VEÍCULOS	26
CAPÍTULO 2 - ESTRATÉGIAS DE SOLUÇÃO EM PROBLEMAS DE ROTEIRIZAÇÃO DE VEÍCULOS: REVISÃO DA LITERATURA	29
2.1 - COMPLEXIDADE COMPUTACIONAL DOS PROBLEMAS DE ROTEIRIZAÇÃO DE VEÍCULOS	29
2.2 - CLASSIFICAÇÃO DAS PRINCIPAIS ESTRATÉGIAS DE SOLUÇÃO EM PROBLEMAS DE ROTEIRIZAÇÃO E PROGRAMAÇÃO DE VEÍCULOS	31
2.2.1 - Classificação das estratégias segundo Bodin e Golden (1981) e Bodin et al. (1983)	
2.2.2 - Classificação segundo Bott e Ballou (1986)	
2.2.3 - Classificação das estratégias de solução segundo Silver, Vidal e Werra (1980), Ball e Magazine (1981) e Zanakis, Evans e Vazacopoulus (1989)	

2.2.4 - Classificação segundo Laporte (1992), Christofides (1985), Osman (1993) e Cunha (1997)	
2.3 - REVISÃO DA LITERATURA DOS PRINCIPAIS MÉTODOS DE SOLUÇÃO PARA PROBLEMAS DE ROTEIRIZAÇÃO DE VEÍCULOS	36
2.3.1 - Problema Clássico de Roteirização de Veículos (<i>Vehicle Routing Problem - VRP</i>)	
2.3.2 - Problema de Roteirização de Veículos com Janelas de Tempo (<i>Vehicle Routing Problem with Time Windows - VRPTW</i>)	
2.3.3 - Problema de Roteirização de Veículos com Janela de Tempo Flexível (<i>Vehicle Routing Problem with Soft Time Windows - VRPSTW</i>)	
2.3.4 - Problema de Roteirização de Veículos com Entregas Fracionadas (<i>Vehicle Routing Problem with Split Deliveries - VRPSD</i>)	
2.3.5 - Problema de Roteirização de Veículos com Janelas de Tempo e Entregas Fracionadas (<i>Vehicle Routing Problem with Time Windows and Split Deliveries - VRPTWSD</i>)	
2.3.6 - Problema de Roteirização de Veículos com Demanda em Arcos, Janelas de Tempo e Entregas Fracionadas (<i>Capacitated Arc Routing Problem with Time Windows and Split Deliveries - CARPTWSD</i>)	
2.3.7 - Problema de Roteirização de Veículos com Frota Heterogênea Fixa (<i>Heterogeneous Fixed Fleet Vehicle Routing Problem - HFFVRP</i>)	
2.3.8 - Problema de Roteirização de Veículos com Frota Heterogênea Fixa e Restrições de Janelas de Tempo (<i>Heterogeneous Fixed Fleet Vehicle Routing Problem with Time Windows - HFFVRPTW</i>)	
2.3.9 - Problema de Dimensionamento e Roteirização de uma Frota Heterogênea de Veículos (<i>Fleet Size and Mix Vehicle Routing Problem - FSMVRP</i>)	
2.3.10 - Problema de Dimensionamento e Roteirização de uma Frota Heterogênea de Veículos com Restrições de Janelas de Tempo (<i>Fleet Size and Mix Vehicle Routing Problem with Time Windows - FSMVRPTW</i>)	
2.4 - CONCLUSÃO DA REVISÃO BIBLIOGRÁFICA E CONTRIBUIÇÕES	80
CAPÍTULO 3 - SCATTER SEARCH E SUAS APLICAÇÕES EM PROBLEMAS DE ROTEIRIZAÇÃO DE VEÍCULOS 81	
3.1 - DESCRIÇÃO DO MÉTODO	83
3.2 - APLICAÇÕES EM PROBLEMAS DE ROTEIRIZAÇÃO DE VEÍCULOS	89
CAPÍTULO 4 - DEFINIÇÃO DO PROBLEMA 91	
4.1 - CLASSIFICAÇÃO SEGUNDO DESROCHERS, LENSTRA E SAVELSBERGH (1990)	92
4.2 - MODELAGEM MATEMÁTICA	94
4.3 - COMPLEXIDADE DO PROBLEMA E ESTRATÉGIA DE SOLUÇÃO	97
CAPÍTULO 5 - ESTUDO DE CASO 98	
5.1 - CENTRO DE DISTRIBUIÇÃO	98
5.2 - TIPOS DE VEÍCULOS	98
5.3 - MODELO DE PREÇO DE FRETE	99
5.4 - DISTÂNCIA	102
5.5 - VELOCIDADES E TEMPO DE DESCARGA	102
5.6 - DEMANDA DAS LOJAS	103

5.7 - JANELAS DE TEMPO DAS LOJAS	104
CAPÍTULO 6 - RESOLUÇÃO DO MODELO	105
6.1 - HEURÍSTICA DE ECONOMIAS (VERSÃO PARALELA)	105
6.1.1 - Frota heterogênea	
6.1.2 - Demanda maior que a capacidade dos veículos	
6.1.3 - Cálculo das economias	
6.1.4 - Restrições de janelas de tempo, capacidade e tipo de veículo	
6.2 - ADAPTAÇÃO DA HEURÍSTICA DE INSERÇÃO SEQUENCIAL I1 DE SOLOMON (AI1)	112
6.3 - ADAPTAÇÃO DA HEURÍSTICA DE INSERÇÃO SEQUENCIAL DE HO E HAUGLAND (AHH)	115
6.4 - ADAPTAÇÃO DA HEURÍSTICA DE INSERÇÃO SEQUENCIAL DE DULLAERT ET AL. (AHD)	115
6.5 - <i>SCATTER SEARCH</i>	118
6.6 - INSTÂNCIAS DE PROBLEMAS DA LITERATURA	141
6.6.1 - Instâncias de problemas de Solomon (1987)	
6.6.2 - Instâncias de problemas de Ho e Haugland (2004)	
6.6.3 - Instâncias de problemas de Liu e Shen (1999)	
CAPÍTULO 7 - ANÁLISE DOS RESULTADOS	152
7.1 - ESTUDO DE CASO	152
7.2 - INSTÂNCIAS DE SOLOMON (1987)	159
7.3 - INSTÂNCIAS DE HO E HAUGLAND (2004)	164
7.4 - INSTÂNCIAS DE LIU E SHEN (1999)	166
CONCLUSÕES E FUTURAS PESQUISAS	172
REFERÊNCIAS BIBLIOGRÁFICAS	174
APÊNDICE	189
A) RESULTADOS DETALHADOS	189
ANEXOS	195
A) EXEMPLO DE APLICAÇÃO DE <i>SCATTER SEARCH</i>	195
B) CÁLCULO DA DISTÂNCIA GEODÉSICA	201
C) INSTÂNCIAS DE PROBLEMAS DE LIU E SHEN (1999)	202

LISTA DE FIGURAS

FIGURA 1.1 - GRAFO DO PROBLEMA HAMILTONIANO	19
FIGURA 2.1 - COMBINANDO DOIS CLIENTES EM UM ROTEIRO	38
FIGURA 2.2 - PASSOS INICIAIS DO ALGORITMO DE ECONOMIAS	38
FIGURA 2.3 - VERSÃO PARALELA DO ALGORITMO DE ECONOMIAS	38
FIGURA 2.4 - VERSÃO SEQUENCIAL DO ALGORITMO DE ECONOMIAS	39
FIGURA 2.5 – INSERÇÃO	62
FIGURA 2.6 - TROCA DE CLIENTES FRACIONADOS	62
FIGURA 2.7 - TROCA DE CLIENTES ENTRE ROTAS	63
FIGURA 2.8 - TROCA DE ARCOS	63
FIGURA 3.1 - PROCEDIMENTO <i>SCATTER SEARCH</i>	85
FIGURA 3.2 - ESQUEMA DO MÉTODO <i>SCATTER SEARCH</i>	86
FIGURA 6.1 - ALGORITMO PARA ATENDIMENTO DAS LOJAS COM DEMANDA SUPERIOR À CAPACIDADE DOS VEÍCULOS	107
FIGURA 6.2 - INTEGRANDO DOIS CLIENTES EM UM ROTEIRO COMPARTILHADO	108
FIGURA 6.3 - ALGORITMO COMPLETO DE ECONOMIAS (VERSÃO PARALELA)	110
FIGURA 6.4 - DESCRIÇÃO COMPLETA DO ALGORITMO AI1	113
FIGURA 6.5 - DESCRIÇÃO COMPLETA DO ALGORITMO AHD	117
FIGURA 6.6 - ALGORITMO DE ELIMINAÇÃO E COMBINAÇÃO DE ROTAS	126
FIGURA 6.7 - ALGORITMO DE INSERÇÃO	127
FIGURA 6.8 - ALGORITMO DE ELIMINAÇÃO E INSERÇÃO PARA INSTÂNCIAS DE SOLOMON (1987)	144
FIGURA 6.9 - ALGORITMO DE INSERÇÃO PARA INSTÂNCIAS DE SOLOMON (1987)	144

LISTA DE QUADROS

QUADRO 1.1 - CLASSIFICAÇÃO DOS PRINCIPAIS PROBLEMAS DE ROTEIRIZAÇÃO DE VEÍCULOS	27
QUADRO 2.1 - CONJUNTO DE PROBLEMAS DE SOLOMON (1987)	49
QUADRO 2.2 – SUMÁRIO DOS PRINCIPAIS MÉTODOS PARA VRP, VRPTW, VRPSD, HFVRP E SUAS EXTENSÕES	72

LISTA DE TABELAS

TABELA 5.1 - CAPACIDADE DOS VEÍCULOS	99
TABELA 5.2 - MODELO DE PREÇO DO FRETE	100
TABELA 5.3 - PREÇOS DOS FRETES	101
TABELA 5.4 - TEMPO DE DESCARGA E VELOCIDADE MÉDIA	103
TABELA 5.5 - DEMANDA DAS LOJAS	103
TABELA 5.6 - JANELAS DE RECEBIMENTO	104
TABELA 6.1 - TIPOS DE VEÍCULOS	121
TABELA 6.2 - CAPACIDADE DOS VEÍCULOS	133
TABELA 6.3 - DEMANDA DOS CLIENTES	133
TABELA 6.4 – MATRIZ DE COMBINAÇÃO	134
TABELA 6.5 - DEMANDA REMANESCENTE	139
TABELA 6.6 - VALORES DE DEMANDA	146
TABELA 7.1 - CUSTO TOTAL DAS SOLUÇÕES	153
TABELA 7.2 - DIFERENÇA PERCENTUAL EM RELAÇÃO À SOLUÇÃO ORIGINAL	154
TABELA 7.3 - NÚMERO DE VEÍCULOS UTILIZADOS	155
TABELA 7.4 - TEMPO DE PROCESSAMENTO	156
TABELA 7.5 - MÉTODO DE DIVERSIFICAÇÃO	157
TABELA 7.6 - RESULTADOS PARA DIFERENTES VALORES DE <i>PSIZE</i>	158
TABELA 7.7 - RESULTADOS DE SCATTER SEARCH PARA DIFERENTES PARÂMETROS	159
TABELA 7.8 - RESULTADOS DE SCATTER SEARCH PARA DIFERENTES VALORES DE b_1, b_2	160
TABELA 7.9 - RESULTADOS DE SCATTER SEARCH PARA AS INSTÂNCIAS DE SOLOMON (1987)	161

TABELA 7.10 - TEMPO MÉDIO DE PROCESSAMENTO (SEGUNDOS) PARA AS CLASSES DE PROBLEMAS DE SOLOMON (1987)	163
TABELA 7.11 - RESULTADOS DE SCATTER SEARCH PARA DIFERENTES VALORES DE b_1 , b_2 , CONSIDERANDO $[L, U] = [0.70, 1.00]$.	164
TABELA 7.12 - COMPARAÇÃO COM OS RESULTADOS DE HO E HAUGLAND (2004) PARA AS DEMANDAS MODIFICADAS DE SOLOMON	165
TABELA 7.13 - TEMPO MÉDIO DE PROCESSAMENTO (SEGUNDOS) PARA AS PARA AS DEMANDAS MODIFICADAS DE HO E HAUGLAND (2004)	166
TABELA 7.14 - RESULTADOS DE SCATTER SEARCH PARA DIFERENTES VALORES DE b_1 , b_2 , CONSIDERANDO O CONJUNTO DE PROBLEMAS DO TIPO A.	167
TABELA 7.15 - COMPARAÇÃO DE SCATTER SEARCH COM OS RESULTADOS DE LIU E SHEN (1999) E DULLAERT ET AL. (2002)	168
TABELA 7.16 - TEMPO MÉDIO DE PROCESSAMENTO (SEGUNDOS) PARA AS CLASSES DE PROBLEMAS DE LIU E SHEN	171

LISTA DE TABELAS - ANEXOS

TABELA 1 - RESULTADOS DETALHADOS	189
TABELA 2 - SOLUÇÕES DIVERSAS GERADAS EM SCATTER SEARCH	195
TABELA 3 - SOLUÇÕES INICIAIS E SOLUÇÕES APÓS MELHORIA	196
TABELA 4 - ITERAÇÕES DO MÉTODO DE MELHORIA	197
TABELA 5 - DISTÂNCIAS ENTRE SOLUÇÕES	198
TABELA 6 - GERAÇÃO DE SUBCONJUNTOS	199
TABELA 7 - CÁLCULO DO SCORE E COMBINAÇÃO DO SUBCONJUNTO	199
TABELA 8 - CONJUNTO DE PROBLEMAS R1 DE LIU E SHEN (1999)	202
TABELA 9 - CONJUNTO DE PROBLEMAS C1 DE LIU E SHEN (1999)	202
TABELA 10 - CONJUNTO DE PROBLEMAS RC1 DE LIU E SHEN (1999)	202
TABELA 11 - CONJUNTO DE PROBLEMAS R2 DE LIU E SHEN (1999)	202

TABELA 12 - CONJUNTO DE PROBLEMAS C2 DE LIU E SHEN (1999) 203

TABELA 13 - CONJUNTO DE PROBLEMAS RC2 DE LIU E SHEN (1999) 203

LISTA DE GRÁFICOS

GRÁFICO 7.1 - CUSTO TOTAL DAS SOLUÇÕES	153
GRÁFICO 7.2 - DIRENÇA PERCENTUAL DOS ALGORITMOS EM RELAÇÃO À SOLUÇÃO ORIGINAL	154
GRÁFICO 7.3 - NÚMERO DE VEÍCULOS UTILIZADOS	155
GRÁFICO 7.4 - TEMPO COMPUTACIONAL DA METAHEURÍSTICA <i>SCATTER SEARCH</i>	156
GRÁFICO 7.5 - DIFERENÇA PERCENTUAL PARA AS CLASSES DE SOLOMON	163
GRÁFICO 7.6 - DIFERENÇA PERCENTUAL PARA AS CLASSES DO CONJUNTO A	169
GRÁFICO 7.7 - DIFERENÇA PERCENTUAL PARA AS CLASSES DO CONJUNTO B	170
GRÁFICO 7.8 - DIFERENÇA PERCENTUAL PARA AS CLASSES DO CONJUNTO C	170

RESUMO

Esta tese estuda a implementação de heurísticas e da metaheurística *scatter search* (SS) em um problema de roteirização de veículos com frota heterogênea, janelas de tempo e entregas fracionadas (*Heterogeneous Fleet Vehicle Routing Problem with Time Windows and Split Deliveries* – HFVRPTWSD). O HFVRPTWSD é uma combinação do problema de roteirização com frota heterogênea (HFVRP), problema de roteirização de veículos com janelas de tempo (VRPTW) e problema de roteirização com entregas fracionadas (VRPSD). O problema é baseado em um único depósito, a demanda dos clientes pode ser maior que a capacidade dos veículos e, além das restrições de janelas de tempo, há também restrições de capacidade dos veículos e restrições quanto ao tipo de veículo.

O VRPSD foi introduzido na literatura por Dror e Trudeau em 1989. No problema de roteirização de veículos com entregas fracionadas, cada cliente pode ser abastecido por mais de um veículo, enquanto no problema clássico de roteirização de veículos (VRP), cada cliente é atendido por um único veículo. Desta forma, para o VRPSD, além dos roteiros de entrega, deve-se determinar a quantidade entregue a cada cliente em cada veículo. Todos os problemas de roteirização com entregas fracionadas encontrados na literatura (VRPSD e suas extensões) têm como característica frota homogênea. O problema estudado neste trabalho difere, portanto, de todos os problemas de roteirização com entregas fracionadas da literatura, pois tem, como característica, frota heterogênea. O mesmo raciocínio vale para problemas de roteirização de veículos com frota heterogênea.

Os modelos são aplicados em uma rede de varejo no Brasil que é abastecida a partir de um centro de distribuição. A rede compõe um total de 519 lojas distribuídas em 12 estados do país. As heurísticas e a metaheurística *scatter search* também são aplicadas em três conjuntos de problemas encontrados na literatura (SOLOMON, 1987; HO E HAUGLAND, 2004; LIU E SHEN, 1999), com o objetivo de avaliar o desempenho dos algoritmos para cada problema.

O problema consiste em determinar, a cada dia, como alocar os caminhões às lojas, a quantidade de carga em cada caminhão a ser entregue em cada uma das lojas, qual o melhor roteiro e o tempo de início de atendimento do primeiro cliente da rota, de forma a minimizar o custo total de distribuição, garantindo que a demanda das lojas seja atendida e as demais restrições do problema sejam respeitadas.

Para a resolução do VRPSD e suas extensões, a única metaheurística encontrada na literatura foi busca tabu. Para o problema de roteirização com frota heterogênea e suas extensões, foram implementadas

apenas as metaheurísticas busca tabu e BATA (*Back-Tracking Adaptative Threshold Accepting*). As estratégias de solução propostas no presente trabalho consistem na implementação de heurísticas construtivas e da metaheurística *scatter search*. As soluções iniciais de SS são obtidas através da implementação de quatro heurísticas construtivas: heurística de economias, heurística de inserção seqüencial baseada nas idéias de Solomon (1987), heurística de inserção seqüencial baseada nas idéias de Ho e Haugland (2004) e adaptação da heurística de inserção seqüencial de Dullaert et al. (2002).

Para o caso real, foi possível uma redução no custo total da frota comparado com a solução atual da empresa. Para algumas instâncias dos três conjuntos de problemas da literatura, os algoritmos apresentaram resultados similares ou superiores às melhores soluções encontradas.

ABSTRACT

This thesis studies the implementation of heuristics and scatter search (SS) metaheuristic in a Heterogeneous Fleet Vehicle Routing Problem with Time Windows and Split Deliveries (HFVRPTWSD). The HFVRPTWSD is a combination of Heterogeneous Fleet Vehicle Routing Problem (HFVRP), Vehicle Routing Problem with Time Windows (VRPTW) and Vehicle Routing Problem with Split Deliveries (VRPSD). The problem is based in a single depot, the demand of each client can be greater than the vehicle's capacity and beyond the time windows constraints, and there are also constraints on the vehicle capacity and vehicles type.

The VRPSD was introduced in the literature by Dror e Trudeau in 1989. In the split deliveries vehicle routing problem, each client can be supplied by more than one vehicle; while in a classic vehicle routing problem (VRP) each client is supplied by only one vehicle. Thus, for the VRPSD, besides the delivery routes, the amount to be delivered to each client in each vehicle must also be determined. All the split delivery vehicle routing problems researched in the literature (VRPSD and its extensions) have as a characteristic the homogeneous fleet. Therefore, the problem studied differs from the split deliveries vehicle routing problems of the literature because it has a heterogeneous fleet. The same reasoning can be applied in heterogeneous fleet vehicle routing problem.

The models will be applied in a retail market in Brazil that is supplied by a distribution center. The market has 519 stores distributed in 12 Brazilian states. The heuristics and the scatter search metaheuristic will also be applied in three benchmark problems (SOLOMON, 1987; HO AND HAUGLAND, 2004; LIU AND SHEN, 1999), aiming to evaluate the design of the algorithms for each problem.

The problem consists in determining, each day, how to allocate the trucks to the stores, the amount to be delivered in each truck to each client, which one is the best route and the initial time for attending the first client, with the aim of minimizing the total distribution cost, attending the clients' demand and respecting all the problem's constraints.

For the VRPSD and its extensions, the only metaheuristic implemented in the literature was tabu search. For the heterogeneous fleet vehicle routing problem and its extensions, only the tabu search and BATA (Back-Tracking Adaptative Threshold Accepting) metaheuristics have been implemented. The strategies proposed here consist in the implementation of constructive heuristics and the scatter search metaheuristic. The initial solutions of SS are obtained with the

implementation of four constructive heuristics: saving heuristics, sequential insertion heuristic based on the ideas of Solomon (1987), sequential insertion heuristic based on the ideas of Ho e Haugland (2004) and adaptation of the sequential insertion heuristic of Dullaert et al. (2002).

For the real case, it was possible to reduce the total fleet cost, when comparing to the actual solution. At some instances of the three benchmark problems, the algorithms presented similar or better results when compared to the best solutions in the literature.

INTRODUÇÃO

O presente trabalho estuda a implementação de novos algoritmos de otimização (heurísticas e metaheurística) em um problema de roteirização de veículos com frota heterogênea, restrições de janelas de tempo e entregas fracionadas (*Heterogeneous Fleet Vehicle Routing Problem with Time Windows and Split Deliveries* – HFVRPTWSD). O HFVRPTWSD é uma combinação do problema de roteirização de veículos com frota heterogênea (HFVRP), com o problema de roteirização de veículos com janelas de tempo (VRPTW) e com o problema de roteirização de veículos com entregas fracionadas (VRPSD).

No problema de roteirização de veículos com entregas fracionadas (VRPSD), cada cliente pode ser abastecido por mais de um veículo, enquanto no problema clássico de roteirização de veículos (VRP) cada cliente é atendido por um único veículo. Desta forma, para o VRPSD, além dos roteiros de entrega, deve-se determinar a quantidade entregue em cada veículo para cada cliente. No HFVRPTWSD estudado nesse trabalho, a demanda dos clientes pode ser maior que a capacidade dos veículos e, além das restrições de janelas de tempo e capacidade dos veículos, há também restrições quanto ao tipo de veículo.

Este capítulo apresenta, primeiramente, a importância dos problemas de roteirização de veículos. Nos itens seguintes, apresenta-se o objetivo do trabalho, a justificativa do tema e contribuições, além do delineamento do trabalho.

1. A IMPORTÂNCIA DOS PROBLEMAS DE ROTEIRIZAÇÃO DE VÉICULOS

No final da década de 1970, significativas mudanças econômicas passaram a afetar as sociedades comercialmente desenvolvidas e industrializadas. De um lado, observou-se um desenvolvimento acentuado da Tecnologia da Informação e de Comunicação, com impactos na gestão empresarial e no mercado financeiro. De outro lado se constata a crescente concorrência entre as empresas que passou a se dar em nível global (NOVAES, p. 322, 2004).

Esta revolução ocorreu não somente no setor industrial, mas também no setor de distribuição, caracterizado pela integração dos membros da cadeia de suprimentos e por

mudanças organizacionais profundas, com influências significativas nos sistemas logísticos das organizações. A distribuição passa a utilizar recursos de Tecnologia da Informação para intercâmbio eletrônico de dados (EDI) de forma mais intensa. As atividades varejistas passam a ter uma maior preocupação com o nível de serviço ao consumidor.

Foi nesse contexto que a logística surgiu como uma das mais importantes competências na gestão de negócios empresariais, capaz de agregar valor ao cliente, tornando-se fonte de vantagem competitiva. A satisfação do cliente passa a ser um elemento fundamental no mercado atual e engloba disponibilidade do produto, agilidade e eficiência na entrega, entre outros elementos, fazendo com que as empresas busquem meios de melhorar e reduzir os custos dos processos logísticos.

Segundo Ballou (2001), dentre todas as atividades envolvidas na cadeia logística, o transporte é aquela que absorve a maior parcela de custos (entre 1/3 e 2/3 do custo logístico total). Um sistema de transporte eficiente e barato contribui para aumentar a concorrência no mercado, elevar as economias de escala de produção e reduzir os preços das mercadorias.

O presente trabalho busca contribuir com a pesquisa de novas estratégias e métodos de solução aplicados a problemas de transporte, através da implementação de heurísticas e metaheurística em um problema de roteirização de veículos.

Os problemas de roteirização de veículos pertencem a uma categoria ampla de problemas de pesquisa operacional conhecida como problemas de otimização de rede. Nessa categoria encontram-se problemas clássicos, como problema de fluxo máximo, problema do caminho mínimo, problema de transporte, problema de designação (GOLDEN; BALL; BODIN, 1981).

Uma das dificuldades de 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 uma visão sistêmica do processo para uma tomada de decisão de qualidade. Deve-se identificar, modelar e classificar corretamente o problema, determinar

suas características mais relevantes e sua relação com outros problemas, para que seja proposta uma estratégia de solução adequada.

2. OBJETIVO

Este trabalho tem como objetivo desenvolver uma nova estratégia de solução para problemas de roteirização de veículos com frota heterogênea, janelas de tempo e entregas fracionadas (*Heterogeneous Fleet Vehicle Routing Problem with Time Windows and Split Deliveries* – HFVRPTWSD). As principais características do problema estudado são:

- Cada cliente pode ser atendido por mais de um veículo (*split deliveries*)
- A demanda de cada cliente pode ser maior que a capacidade dos veículos
- Frota heterogênea e ilimitada
- É baseado em um único depósito

O problema apresenta as seguintes restrições:

- Janelas de tempo
- Capacidade dos veículos
- Restrição quanto ao tipo de veículo para algumas lojas

As estratégias de solução propostas consistem na implementação de heurísticas construtivas e da metaheurística *scatter search*. As soluções iniciais são obtidas através da implementação de quatro heurísticas construtivas: heurística de economias, heurística de inserção seqüencial baseada nas idéias de Solomon (1987), heurística de inserção seqüencial baseada nas idéias de Ho e Haugland (2004) e adaptação da heurística de inserção seqüencial de Dullaert et al. (2002). Como o número de veículos é ilimitado, o algoritmo também é capaz de resolver o problema de dimensionamento e roteirização de uma frota heterogênea de veículos com janelas de tempo e entregas fracionadas (*Fleet Size and Mix Vehicle Routing with Time Windows and Split Deliveries*).

O problema consiste em determinar a cada dia, como alocar os caminhões às lojas, a quantidade de carga em cada caminhão a ser entregue para cada uma das lojas, qual o melhor roteiro e o tempo de início de atendimento do primeiro cliente da rota, de forma a minimizar o custo total de distribuição e o tempo total de viagem, garantindo que a demanda das lojas será atendida e as restrições do problema serão respeitadas.

Os modelos são aplicados em uma rede de varejo no Brasil que é abastecida a partir de um centro de distribuição. A rede compõe um total de 519 lojas distribuídas em 12 estados do país. As heurísticas e a metaheurística também são aplicadas em três conjuntos de problemas encontrados na literatura (SOLOMON, 1987; HO; HAUGLAND, 2004; LIU; SHEN, 1999), com o objetivo de avaliar o desempenho dos algoritmos para cada tipo de problema.

3. JUSTIFICATIVA DO TEMA E CONTRIBUIÇÕES

O VRPSD foi introduzido na literatura por Dror e Trudeau em 1989, após outros problemas de roteirização de veículos. Ainda há, portanto, uma grande área a ser explorada, justificando o estudo de novos métodos de solução para este problema e suas extensões. Todos os problemas de roteirização de veículos com entregas fracionadas encontrados na literatura e suas extensões (DROR; TRUDEAU, 1989, 1990; FRIZZELL; GIFFIN, 1992; DROR et al., 1994; FRIZZELL; GIFFIN, 1995; MULLASERIL et al., 1997; BELENGUER et al., 2000; ARCHETTI et al., 2003a; ARCHETTI et al., 2003b; HO; HAUGLAND, 2004) têm como característica frota homogênea. Na maioria deles, a demanda de cada cliente é menor que a capacidade dos veículos e não há restrição de janela de tempo. Portanto, o problema estudado neste trabalho difere da maioria dos problemas de roteirização com entregas fracionadas, pois tem como características frota heterogênea, a demanda de cada cliente pode ser maior que a capacidade dos veículos e há restrições de janelas de tempo.

Segundo Gendreau et al. (1999) e Tarantilis et al. (2004), os problemas de roteirização de veículos com frota heterogênea são divididos em duas categorias: problema de roteirização com frota heterogênea fixa (*Heterogeneous Fixed Fleet Vehicle Routing Problem* – HFFVRP), no qual a frota de veículos é limitada e o objetivo é determinar os roteiros de entrega, e problema de dimensionamento e roteirização de uma frota heterogênea de veículos (*Fleet Size and Mix Vehicle Routing Problem* – FSMVRP), no qual a frota de veículos é ilimitada e, além dos roteiros, busca-se determinar a configuração ideal de veículos. A

maioria das estratégias de solução propostas para problemas de roteirização com frota heterogênea (e suas extensões) encontradas na literatura pertence à classe FSMVRP (GOLDEN et al., 1984; DESROCHERS; VERHOOG, 1991; GOUVÊA, 1992; SALHI; HANDI, 1993; GENDREAU et al., 1999; TEIXEIRA; CUNHA, 2002; WASSAN; OSMAN, 2002). Além do FSMVRP, podemos encontrar estratégias para o HFFVRP (TAILLARD, 1999; TARANTILIS et al., 2004), HFFVRP com janelas de tempo (ROCHAT; SEMET, 1984; CUNHA, 1997) e FSMVRP com janelas de tempo (LIU; SHEN, 1999; DULLAERT et al., 2002).

Conclui-se, portanto, que o HFVRPTWSD combinado nunca foi estudado na literatura. Como o problema estudado é de grande complexidade computacional (*NP-completo*), torna-se inviável a obtenção da solução ótima, justificando a utilização de heurísticas e metaheurísticas para resolução do problema.

A maioria das aplicações da metaheurística *scatter search* surgiu na década de 90, e poucos estudos foram feitos em problemas de roteirização de veículos (REGO; LEÃO, 2000; CORBERÁN et al., 2002; ALEGRE; LAGUNA; PACHECO, 2004; CHU; LABADI; PRINS, 2006; RUSSELL; CHIANG, 2006), nenhum deles em problemas de roteirização de veículos com entregas fracionadas (e suas extensões) e nem em problemas de roteirização de veículos com frota heterogênea (e suas extensões). A única metaheurística implementada para problemas de roteirização com entregas fracionadas foi busca tabu (ARCHETTI et al., 2003b; HO; HAUGLAND, 2004) e para problemas com frota heterogênea (e suas extensões) foi busca tabu (ROCHAT; SEMET, 1994; GENDREAU et al., 1999; WASSAN; OSMAN, 2002) e BATA – *Back-Tracking Adaptative Threshold Accepting* (TARANTILIS et al., 2004).

4. DELINEAMENTO DO TRABALHO

O capítulo 1 apresenta a taxonomia dos principais problemas de roteirização de veículos encontrados na literatura e mostra como o HFVRPTWSD, foco do trabalho, está classificado neste contexto.

Já o capítulo 2 faz uma revisão da literatura das principais estratégias de solução implementadas em problemas de roteirização de veículos. Este capítulo apresenta o estado-da-arte da modelagem de problemas de roteirização e programação de veículos.

O capítulo 3 descreve a metaheurística *scatter search* e suas aplicações em problemas de roteirização de veículo.

O capítulo 4 apresenta a definição do problema proposto, descrevendo a classificação, formulação matemática e complexidade do problema.

O capítulo 5 apresenta o estudo de caso implementado neste trabalho, descrevendo todos os dados de entrada do modelo.

O capítulo 6 descreve a implementação das heurísticas construtivas e da metaheurística *scatter search* para a resolução do modelo proposto.

No capítulo 7 são apresentadas a análise e interpretação dos resultados e, a seguir, estão as considerações finais, conclusões e futuras pesquisas.

CAPÍTULO 1

TAXONOMIA DOS PROBLEMAS DE ROTEIRIZAÇÃO DE VEÍCULOS

Primeiramente são apresentados os conceitos básicos e os principais parâmetros que caracterizam um problema de roteirização de veículos. A partir do problema básico de roteirização de veículos são apresentadas suas extensões, de acordo com os seguintes parâmetros: função objetivo, restrições, variáveis de decisão e hipóteses.

A seguir é apresentada a classificação dos problemas de roteirização de veículos segundo Bodin et al. (1983), Ronen (1988), Assad (1988) e Desrochers, Lenstra e Savelsbergh (1990).

Este capítulo também descreve os problemas clássicos existentes de roteirização de veículos e mostra como o HFVRPTWSD está inserido nesta classificação.

1.1 O PROBLEMA CLÁSSICO DE ROTEIRIZAÇÃO DE VEÍCULOS E SUAS EXTENSÕES

Christofides (1985) define o problema básico de roteirização de veículos como o problema de distribuição no qual veículos localizados em um depósito central devem ser programados para visitar clientes geograficamente dispersos, de modo a atender suas demandas conhecidas. As restrições mais usuais do problema estão relacionadas às capacidades dos veículos.

O problema básico de roteirização ignora um grande número e variedade de restrições adicionais e extensões, que são freqüentemente encontradas em situações reais.

Tendo em vista o grande número de situações práticas que dão origem aos problemas de roteirização de veículos, Cristofides (1985) apresenta o problema básico de roteirização de veículos e o descreve como a essência de todos os problemas de roteirização de veículos.

Os problemas de roteirização de veículos podem ser classificados em diversas categorias e tipos. Os vários problemas diferem entre si em aspectos relacionados ao tipo de operação, ao tipo de carga, ao tipo de frota utilizada, à localização dos clientes, ao tipo de restrições, ao tipo de função objetivo e vários outros fatores que serão relacionados adiante.

A seguir são apresentadas as extensões do problema básico de roteirização de veículos, baseadas nos parâmetros mais relevantes e comuns. Os principais parâmetros que caracterizam um problema de roteirização foram obtidos a partir dos trabalhos de Christofides (1985), Bodin e Golden (1981), Bodin et al. (1983), Assad (1988), Ronen (1988) e Brejon (1998). Neste trabalho os parâmetros são classificados em 4 categorias: função objetivo, restrições, variáveis de decisão e hipóteses/recursos e características do problema.

Função objetivo

- minimizar os custos totais de distribuição, que incluem os custos fixos (custos de capital do veículo, salários de motoristas e ajudantes, e outras despesas eventuais como licenciamento, seguros, taxas, etc.) e os custos variáveis (custos do veículo que variam conforme a distância);
- minimizar a distância total percorrida;
- minimizar a duração das rotas;
- minimizar o número de veículos;
- maximizar a função utilidade baseada no nível de serviço e/ou prioridades dos clientes.

As principais restrições estão descritas abaixo.

Restrições dos veículos

- limite de capacidade dos veículos;
- limite com relação ao tipo de carga dos veículos;
- operação de carga e descarga dos veículos;
- número e tipo de veículos disponíveis.

Restrições junto aos clientes

- janelas de tempo dos clientes (intervalo de atendimento de cada cliente);

- atendimento total ou parcial das demandas;
- prioridade no atendimento de clientes;
- tempo máximo permitido para carga e descarga;
- necessidade ou restrição de serviço em algum dia específico da semana.

Restrições das rotas

- horários de início e término das viagens;
- tempo máximo de viagem de um veículo;
- distância máxima percorrida;
- locais de paradas fixas, etc.

As variáveis de decisão encontram-se abaixo.

Variáveis de decisão

- roteiro a ser percorrido por cada veículo;
- qual veículo é designado para cada cliente;
- qual a quantidade de carga transportada para cada cliente da rota;
- tempo de início de atendimento do primeiro cliente da rota.

A seguir estão detalhadas as principais hipóteses/recursos e características do problema.

Tipo de operação

- coleta;
- entrega;
- coleta e entrega simultaneamente;
- coleta (ou entrega) com carga de retorno.

Tipo de carga

- única ou carga de lotação;
- múltiplas cargas ou carga fracionada.

Tipo de demanda

- determinística;

- estocástica.

Localização da demanda

- demanda localizada somente em arcos;
- demanda localizada somente em nós;
- demanda localizada em arcos e nós.

Tamanho da frota

- limitada;
- ilimitada.

Tipo de frota

- homogênea;
- heterogênea.

Depósitos e localização de veículos

- um único depósito;
- vários depósitos;
- quantidade de produtos disponíveis no depósito central para entrega aos clientes;
- número de bases de origem e destino dos veículos.

Jornada de trabalho

- duração;
- horário de almoço e outras interrupções;
- permissão para viagens com mais de um dia de duração;
- número de tripulantes por veículo.

Pagamento dos tripulantes

- por jornada de trabalho;
- por produtividade;
- jornada e horas extras.

Estrutura da rede

- direcionada;

- não direcionada;
- mista;
- euclidiana.

Horizonte de planejamento

- curto prazo;
- longo prazo.

Outras hipóteses

- cada veículo pode visitar um cliente uma única vez durante a rota;
- um cliente pertence a uma única rota;
- um cliente pode pertencer a mais de uma rota;
- quando o veículo visita um cliente da rota todos os clientes são visitados, etc.

1.2 CLASSIFICAÇÃO DOS PROBLEMAS DE ROTEIRIZAÇÃO DE VEÍCULOS

Conforme dito no item 1.1 uma das dificuldades de se modelar e resolver um problema de roteirização advém da grande quantidade de parâmetros que podem influenciar este tipo de problema. Uma visão sistêmica dos principais problemas de roteirização de veículos e uma adequada classificação, identificando os aspectos mais relevantes, permite que seja implementada uma estratégia de solução adequada.

Neste item 1.2 são descritas classificações de problemas de roteirização de veículos segundo Bodin et al. (1983), Ronen (1988), Assad (1988) e Desrochers, Lenstra e Savelsbergh (1990). A descrição dos problemas clássicos de roteirização de veículos, de acordo com as classificações de Bodin et al. (1983) e Solomon e Desrosiers (1988), será descrita no item 1.3.

1.2.1 CLASSIFICAÇÃO SEGUNDO BODIN ET AL. (1983)

Bodin et al. (1983) propõem uma estrutura que classifica os problemas em função de restrições de aspectos espaciais e/ou temporais. Assim, os problemas de roteirização podem ser classificados em três grupos: problema de roteirização pura, problemas de programação de veículos e problemas combinados de roteirização e programação.

Problema de roteirização pura: não há restrições temporais, relacionadas ao horário de atendimento dos clientes, nem relações de precedência entre os clientes. Neste tipo de problema consideram-se apenas aspectos espaciais, e o objetivo é construir um conjunto de roteiros viáveis com o menor custo possível.

Problema de programação de veículos: há restrições de horários pré-estabelecidos para cada atividade a ser executada, como horário de chegada e saída das lojas, horário de saída do depósito, parada para reabastecimento, etc. Neste tipo de problema consideram-se tanto os aspectos espaciais quanto os temporais.

Problemas Combinados de Roteirização e Programação de veículos: há restrições de precedência entre tarefas e/ou restrições de janela de tempo. Relações de precedência ocorrem, por exemplo, quando a entrega de uma mercadoria deve ser precedida pela sua coleta. As janelas de tempo correspondem ao intervalo de tempo em que pode ocorrer o início do atendimento dos clientes. Segundo Bodin e Golden (1981), os problemas que ocorrem na prática normalmente estão nessa 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.

A taxonomia dos problemas de roteirização de veículos proposta no presente trabalho, que será apresentada no item 1.4 (quadro 1.1), é uma extensão dessa classificação.

1.2.2 CLASSIFICAÇÃO SEGUNDO RONEN (1988)

Ronen (1988) apresenta uma classificação baseada em três classes principais que diferem entre si em função do ambiente operacional e dos objetivos, sendo que em cada uma das classes pode haver uma variedade de considerações:

Problemas relacionados ao transporte de passageiros: correspondem aos sistemas de transporte de ônibus, táxi, pessoas, idosos (*dial-a ride*) e transportes escolares de ônibus.

Problemas de prestação de serviços: roteirização e programação de serviços como coleta de lixo, entrega postal, varrição de ruas e leitura de parquímetros, entre outros;

Problemas de roteirização e programação ou transporte de carga, que é o objetivo do trabalho.

Os critérios de classificação para os problemas de transporte de carga são:

- 1) Tamanho da frota
- 2) Composição da frota
- 3) Estrutura de custos de frota
- 4) Componentes de custo
- 5) Número de bases de origem e destino dos veículos
- 6) Natureza da demanda
- 7) Tipo de operação
- 8) Número de viagens por veículo num determinado período
- 9) Duração ou tempo máximo permitido de viagem
- 10) Tipo de entrega
- 11) Distâncias e tempos
- 12) Função objetivo do modelo

1.2.3 CLASSIFICAÇÃO SEGUNDO ASSAD (1988)

Assad (1988) alega que a maior dificuldade em encontrar um esquema de classificação apropriado está em considerar ou não as restrições do problema e o método de solução proposto. O autor sugere que cada problema prático seja caracterizado particularmente, de acordo com um conjunto de elementos. Outra possível classificação baseia-se no tempo em que as informações das demandas estão disponíveis. Nos problemas clássicos de roteirização, assume-se que a demanda é conhecida antecipadamente (demanda determinística). Na roteirização dinâmica a demanda é estocástica, ou seja, os roteiros ocorrem em tempo real.

O conjunto de elementos proposto por Assad (1988) para caracterização dos problemas de roteirização é:

- Natureza das demandas: somente coletas ou entregas; *backhauls*; um ou múltiplos produtos; atendimento total ou parcial da demanda; prioridade no atendimento aos clientes; terceirização ou não;
- Características da demanda: determinística ou estocástica, constante ou variável, etc;
- 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; tempos de viagem; localização dos veículos.

1.2.4 CLASSIFICAÇÃO SEGUNDO DESROCHERS, LENSTRA E SAVELSBERGH (1990)

Desrochers, Lenstra e Savelsbergh (1990) propuseram um esquema muito elegante que, além de servir para classificar uma grande variedade de problemas de roteirização e programação de veículos, dá suporte ao desenvolvimento de modelos para sistemas desta área. A idéia é dar diretrizes para uma representação teórica do problema real, servindo como base para o desenvolvimento de modelos e sistemas, e possibilitar uma escolha apropriada do algoritmo utilizado. Os principais parâmetros abordados neste esquema estão classificados em quatro categorias:

CLASSIFICAÇÃO:

- ENDEREÇO
- VEÍCULO

- CARACTERÍSTICAS DO PROBLEMA
- OBJETIVOS

O primeiro campo ENDEREÇO define as características que podem ser associadas a um único endereço. Os endereços deverão estar localizados sobre uma rede $G = (V, E)$ com um conjunto V de nós e um conjunto E (não direcionado) de arestas e de arcos (direcionado). É composto de quatro subcampos:

ENDEREÇOS

- número de depósitos
- tipo de demanda
- restrições de programação dos endereços
- restrição de seleção dos endereços

O primeiro subcampo corresponde ao *número de depósitos*. Existem problemas com um único depósito e problemas com múltiplos depósitos.

O segundo subcampo especifica o *tipo de demanda* e está dividido em três partes. A primeira parte corresponde à localização da demanda. O cliente pode estar localizado sobre o nó, nos arcos da rede, tanto nos nós quanto nos arcos, ou estar associado a um par origem-destino. A segunda parte especifica se todas as demandas são do mesmo tipo (só entregas ou só coletas) ou não (mistura de entregas e coletas). A terceira parte especifica a natureza da demanda, ou seja, se é determinística ou estocástica.

O terceiro subcampo do endereço especifica *restrições de programação dos endereços*, isto é, os aspectos temporais da demanda. Para cada demanda, podem ou não existir restrições temporais.

O último subcampo especifica a *restrição de seleção dos endereços*. Há duas classes de problemas. A primeira classe é dividida em três subclasses: todos os endereços devem ser visitados; um dado subconjunto de endereços deve ser visitado e os outros são visitados se for vantajoso; ou os endereços são particionados em subconjuntos e pelo menos um endereço em cada subconjunto deve ser visitado. Na segunda classe, um número de planos é construído

para um certo período de tempo, durante o qual os endereços devem ser visitados com uma dada prioridade ou uma dada freqüência.

O segundo campo VEÍCULO define as características do veículo em suas rotas. Há três tipos de informações neste campo: número de veículos, características físicas do veículo e restrições temporais sobre uma rota. É dividido em cinco subcampos:

VEÍCULOS

- número de veículos
- restrição de capacidade
- restrição de mercadoria
- restrição de disponibilidade do veículo
- restrição de tempo de duração da rota

O primeiro subcampo especifica o *número de veículos*. O segundo campo especifica características físicas de capacidade. A frota pode ser homogênea (todos os veículos têm a mesma capacidade) ou heterogênea (veículos de diferentes capacidades). O terceiro subcampo especifica características físicas da presença de compartimentos. Alguns veículos têm compartimentos intercambiáveis, podendo ser usados para separar mercadorias incompatíveis. Outros veículos possuem compartimentos especiais, que são utilizados para armazenar um tipo especial de mercadoria como, por exemplo, comidas congeladas.

O quarto e quinto subcampos especificam restrições temporais, como intervalos de disponibilidade dos veículos e os limites inferiores e superiores de duração das rotas.

O terceiro campo CARACTERÍSTICAS DO PROBLEMA define o tipo de rede utilizada, a estratégia de serviço, e as restrições nas relações entre os endereços e os veículos. É dividido em cinco subcampos:

CARACTERÍSTICAS DO PROBLEMA

- Tipo de rede
- Tipo de estratégia de serviço
- Restrição endereço-endereço
- Restrição endereço-veículo

- Restrição veículo-veículo

O primeiro subcampo especifica as *propriedades da rede* (direcionada, não direcionada ou mista) e dos tempos de viagem.

O segundo subcampo especifica a *estratégia de serviço* adotada e compõe quatro estratégias:

- 1) A primeira estratégia permite ou não a quebra da demanda.
- 2) As entregas podem ser diretas (*backhaul*) ou fracionadas. A roteirização pode envolver coletas e entregas ou somente entrega.
- 3) Na maioria dos casos um veículo realiza no máximo uma rota por período, mas é possível permitir mais que uma rota por veículo.
- 4) Geralmente os veículos saem e retornam ao mesmo depósito, mas isto pode ser relaxado de forma a permitir rotas multi-depósitos.

Os outros subcampos especificam relações entre dois endereços, entre um endereço e um veículo, ou entre dois veículos.

No caso de *restrições entre endereços*, um exemplo pode ser restrições de precedência entre dois clientes, no qual o veículo deve visitar um cliente antes de visitar o outro. Pode ocorrer também que um endereço a ser visitado por um veículo deva pertencer à mesma rota que outro endereço específico.

Uma *restrição entre endereços e veículos* pode ocorrer quando determinados clientes não podem receber determinado veículo, devido a restrições de descarga do cliente e/ou tipo de carga que o veículo pode levar. Podem ocorrer também restrições entre depósitos e veículos (tempos de carregamento).

O último tipo de restrição é a *sincronização dos veículos*, que ocorre quando dois ou mais veículos devem trocar cargas ou assistir um ao outro.

O quarto campo define a função objetivo. Para especificar uma função objetivo são introduzidos cinco elementos quantitativos:

FUNÇÃO OBJETIVO

- Duração total da rota
- Função custo do veículo
- Função custo do endereço
- Função penalidade do veículo
- Função penalidade de endereço

Uma função custo do veículo pode ser utilizada para modelar o tamanho e tipo de frota. Uma função custo no endereço leva em consideração o custo incorrido ao não se garantir o nível de serviço pretendido. As funções penalidades permitem modelar os custos incorridos com a violação de restrições flexíveis. As restrições de horas extras de motoristas, por exemplo, podem ser violadas através de penalidades. As restrições que são consideradas flexíveis são listadas como argumentos das funções penalidades do veículo e endereço. Na prática os problemas têm geralmente uma função objetivo composta.

A classificação do problema estudado neste trabalho, de acordo com Desrochers, Lenstra e Savelsbergh (1990), está descrita no capítulo 4 e tem como objetivo apresentar, de forma clara e detalhada, os principais parâmetros do problema.

1.3 PROBLEMAS CLÁSSICOS DE ROTEIRIZAÇÃO DE VEÍCULOS

Este item apresenta os problemas clássicos de roteirização e programação de veículos com base nos trabalhos de Bodin et al. (1983) e Solomon e Desrosiers (1988).

• Problema do Caixeiro Viajante (*Traveling Salesman Problem – TSP*)

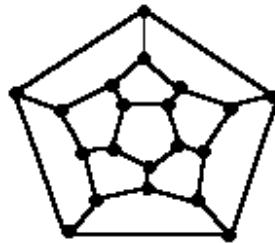
O problema do caixeiro viajante (TSP) é um problema de otimização associada à determinação dos caminhos denominados hamiltonianos. Sua origem advém de Willian Rowan Hamilton que propôs um jogo cujo desafio consistia em encontrar uma rota através dos vértices de um dodecaedro de tal modo que a rota iniciasse e terminasse no mesmo vértice, sem nunca repetir uma visita. Assim, o objetivo do TSP é encontrar em um grafo

$G = (N, A)$ o caminho hamiltoniano de menor custo, de forma que todos os vértices sejam visitados uma única vez.

O problema consiste em determinar um único roteiro com menor custo possível que permita ao caixeiro viajante (veículo) visitar todos os nós (clientes) de uma rede, uma única vez. O problema é baseado em um único depósito e o veículo deve sair e retornar à mesma base. Neste problema não há restrições de capacidade de veículos, e a demanda é determinística.

O grafo do problema e uma das possíveis soluções do jogo são apresentados na figura abaixo.

Figura 1.1 - Grafo do problema hamiltoniano



Fonte: Goldbarg e Luna (2000)

Existem diversas formulações para este problema. Dantzig, Fulkerson e Johnson (1954) formularam o TSP como um problema de programação binária sobre um grafo $G = (N, A)$ da seguinte forma:

$$\text{minimizar } z = \sum_{j=1}^n \sum_{i=1}^n c_{ij} x_{ij}$$

sujeito a:

$$\sum_{i=1}^n x_{ij} = 1 \quad \forall j \in N \quad (1.1)$$

$$\sum_{j=1}^n x_{ij} = 1 \quad \forall i \in N \quad (1.2)$$

$$\sum_{i,j \in S} x_{ij} \leq |S| - 1 \quad \forall S \subset N \quad (1.3)$$

$$x_{ij} \in \{0, 1\} \quad \forall i, j \in N \quad (1.4)$$

A variável binária x_{ij} assume valor igual a 1, se o arco $(i, j) \in A$ for escolhido para integrar a solução, e 0 em caso contrário. Já c_{ij} corresponde ao custo associado ao arco (i, j) , e S é um subgrafo de G em que $|S|$ representa o número de vértices desse subgrafo. As restrições (1.1) e (1.2) garantem que cada vértice seja visitado uma única vez. O conjunto de restrições (1.3) impede que haja formação de *subtours*.

- **Problema do Carteiro Chinês (*Chinese Postman Problem – CPP*)**

O problema do carteiro chinês (CPP) é uma variação do problema do caixeiro viajante, porém, ao invés de nós, os clientes estão localizados em arcos. O problema consiste em determinar um único roteiro com menor custo possível que permita ao carteiro chinês (veículo) visitar todos os arcos (clientes) de uma rede, uma única vez. O problema é baseado em um único depósito e o veículo deve sair e retornar à mesma base. Neste problema, também não há restrições de capacidade de veículos e a demanda é determinística.

- **Problema de Múltiplos Caixeiros Viajantes (*The Multiple Traveling Salesman Problem – MTSP*)**

O problema de múltiplos caixeiros viajantes (MTSP) é uma extensão do problema do caixeiro viajante, porém, ao invés de um único roteiro, determinam-se múltiplos roteiros. O problema consiste em determinar múltiplos roteiros com menor custo possível, de forma que cada caixeiro viajante deve visitar pelo menos um nó da rede, e cada nó deve ser visitado uma única vez. O problema é baseado em um único depósito e o veículo deve sair e retornar à mesma base. Neste problema não há restrições de capacidade de veículos e a demanda é determinística.

- **Problema Clássico de Roteirização de Veículos (*Vehicle Routing Problem – VRP*)**

O problema de roteirização de veículos (VRP) tem como objetivo encontrar um conjunto de rotas com o menor custo possível (minimizar custo total de viagem, distância total percorrida, etc), iniciando e terminando no depósito, de forma que a demanda de todos

os nós são atendidas. A demanda é determinística. Esse problema é uma extensão do problema de múltiplos caixeiros viajantes, onde se acrescenta a restrição de capacidade dos veículos. Algumas formulações também apresentam restrição de tempo máximo de viagem.

Uma formulação básica do problema de roteirização de veículos é apresentada por Fisher e Jaikumar e está detalhada a seguir:

Parâmetros

n = número total de clientes a serem atendidos, que varia de 1 a n e 0 representa o depósito

NV = número total de veículos disponíveis

C_v = Capacidade máxima do veículo v (peso ou volume)

d_i = demanda do cliente i

c_{ij} = custo de viagem do cliente i ao cliente j

Variáveis de decisão

$$x_{ij}^v = \begin{cases} 1, & \text{se o arco de } i \text{ até } j \text{ é percorrido pelo veículo } v \\ 0, & \text{caso contrário} \end{cases}$$

$$y_i^v = \begin{cases} 1, & \text{se o pedido do cliente } i \text{ for entregue pelo veículo } v \\ 0, & \text{caso contrário} \end{cases}$$

Formulação do modelo

$$\min \sum_i \sum_j \sum_v c_{ij} x_{ij}^v$$

s.a.

$$\sum_v y_i^v = 1, \quad i = 1, \dots, n \quad (1.5)$$

$$\sum_v y_i^v = NV, \quad i = 0 \quad (1.6)$$

$$\sum_i d_i y_i^v \leq C_v, \quad v = 1, \dots, NV \quad (1.7)$$

$$\sum_i x_{ij}^v = y_j^v, \quad j = 0, \dots, n, \quad v = 1, \dots, NV \quad (1.8)$$

$$\sum_j x_{ij}^v = y_i^v, \quad i = 0, \dots, n, \quad v = 1, \dots, NV \quad (1.9)$$

$$\sum_{ij \in S} x_{ij}^v = x_{ij}^v \leq |S| - 1, \quad S \subseteq \{1, \dots, n\}, \quad 2 \leq |S| \leq n - 1, \quad v = 1, \dots, NV \quad (1.10)$$

$$x_{ij}^v = 0 \text{ ou } 1, \quad i = 0, \dots, n \quad j = 0, \dots, n, \quad v = 1, \dots, NV \quad (1.11)$$

$$y_i^v = 0 \text{ ou } 1, \quad i = 0, \dots, n, \quad v = 1, \dots, NV \quad (1.12)$$

O objetivo do modelo é minimizar o custo total de viagem.

A restrição (1.5) garante que cada ponto (cliente) seja visitado por apenas um veículo. Já a restrição (1.6) garante que todas as rotas comecem e terminem no depósito ($i = 0$).

A restrição (1.7) garante que a capacidade dos veículos não será excedida.

As restrições (1.8) e (1.9) garantem que os veículos não interrompam suas rotas em um cliente. São as restrições de conservação dos fluxos de entrada e saída.

A restrição (1.10) garante que não sejam formadas sub-rotas.

As restrições (1.11) e (1.12) garantem que as variáveis x_{ij}^v e y_i^v sejam binárias.

- **Problema de Roteirização de Veículos com Múltiplos Depósitos (*Multi-Depot Vehicle Routing Problem – MDVRP*)**

O problema de roteirização de veículos com múltiplos depósitos (MDVRP) é uma generalização do problema clássico de roteirização, onde, ao invés de um único depósito, há múltiplos depósitos. Os veículos devem sair e retornar a um depósito, entre os depósitos existentes.

- **Problema de Roteirização de Veículos com Demanda em Arcos (*Capacitated Arc Routing Problem – CARP*)**

O CARP é uma extensão do problema do carteiro chinês, acrescentando-se a restrição de capacidade de veículos, e é também uma variação do problema clássico de roteirização de veículos, no qual os clientes estão localizados em arcos ao invés de nós.

- **Problema de Roteirização de Veículos com Demanda Estocástica (*Stochastic Vehicle Routing Problem – SVRP*)**

Este problema é uma variação do problema clássico de roteirização de veículos, onde a demanda passa a ser estocástica ao invés de determinística, como ocorre nos demais problemas.

- **Problema de Roteirização de Veículos com Entregas Fracionadas (*Vehicle Routing Problem with Split Deliveries – VRPSD*)**

O VRPSD é uma variação do problema clássico de roteirização de veículos, onde, ao invés de uma única entrega, cada ponto de demanda poder ser abastecido por mais de um veículo. A demanda de cada ponto pode ser maior ou menor que a capacidade do veículo.

- **Problema de Dimensionamento e Roteirização de uma Frota Homogênea de Veículos (*Fleet Size and Vehicle Routing Problem – FSVRP*)**

No 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. Neste problema, a frota de veículos é homogênea, portanto, a capacidade e os custos são idênticos.

- **Problema de Roteirização de Veículos com Frota Heterogênea Fixa (*Heterogeneous Fixed Fleet Vehicle Routing Problem – HFFVRP*)**

O HFFVRP ou, simplesmente, problema de roteirização de veículos com frota heterogênea (*Heterogeneous Fleet Vehicle Routing Problem – HFVRP*) é uma variação do problema clássico de roteirização de veículos, no qual, ao invés de homogênea, a frota de veículos é heterogênea. Neste problema, o número de veículos de cada tipo é limitado (fixo). O objetivo é minimizar a soma dos custos fixos e dos custos variáveis que podem ser dependentes ou não do tipo de veículo.

- **Problema de Dimensionamento e Roteirização de uma Frota de Veículos Heterogênea (*Fleet Size and Mix Vehicle Routing Problem – FSMVRP*)**

Este problema é uma generalização do FSVRP, porém, difere no tipo de frota de veículos que, neste caso, é heterogênea. O FSMVRP é também uma variação do problema anterior, onde, ao invés de limitada, a frota de veículos é 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.

- **Problema de Roteirização de Veículos Multi-Período ou Periódico (*Periodic Vehicle Routing Problem – PVRP*)**

Este problema é uma extensão do problema clássico de roteirização de veículos, no qual o horizonte de tempo passa a ser M dias ao invés de um único dia.

- **Problema de Roteirização de Veículos com Tempo Dependente (*Time Dependent Vehicle Routing Problem – TDVRP*)**

Este problema é uma variação do problema clássico de roteirização de veículos, no qual o tempo de viagem entre dois clientes ou entre um cliente e o depósito depende, além da distância entre os pontos, do horário do dia.

- **Problema de Roteirização (e Programação) de Veículos com Janelas de Tempo (*Vehicle Routing (and Scheduling) Problem with Time Windows – VRSPTW ou VRPTW*)**

O problema de roteirização (e programação) de veículos com janela de tempo é uma generalização do problema clássico de roteirização de veículos (VRP), adicionando restrições de janela de tempo.

- **Problema de Roteirização (e Programação) de Veículos com Janela de Tempo Flexível (*Vehicle Routing (and Scheduling) Problem with “Soft” Time Windows – VRSPSTW ou VRPSTW*)**

O problema de roteirização de veículos com janela de tempo flexível (VRPSTW) é semelhante ao problema anterior, porém, neste caso, as janelas de tempo podem ser violadas mediante o pagamento de penalidades.

- **Problema de Coleta e Entrega (*Pickup and Delivery Problems – PDP*) e *Dial-a-ride Problem (DARP)***

O PDP é uma extensão do problema de roteirização de veículos (VRP), acrescentando relações de precedência entre os clientes, no qual a tarefa de coleta de um cliente deve preceder a de entrega no ponto de destino, o que inviabiliza o uso de algumas heurísticas mais gerais (CUNHA, 2003). As cargas são transportadas do depósito aos clientes e entre os clientes, resultando na relação de precedência.

O problema *dial-a-ride* (DARP) é um PDP em que as cargas a serem transportadas são pessoas e são iguais a um. O conceito *dial-a-ride* surgiu de uma aplicação típica, onde os clientes telefonam, requisitando a visita de um veículo em determinados pontos de origem e destino, em horários correspondentes.

O problema estudado neste trabalho é uma combinação dos problemas de roteirização de veículos com frota heterogênea (HFVRP) e ilimitada, dos problemas de roteirização com janelas de tempo (VRPTW) e dos problemas de roteirização de veículos com entregas fracionadas (VRPSD). Como o número de veículos é ilimitado, o algoritmo proposto também é capaz de resolver problemas de dimensionamento e roteirização de uma frota heterogênea de veículos com janelas de tempo e entregas fracionadas (*Fleet Size and Mix Vehicle Routing Problem with Time Windows and Split Deliveries - FSMVRPTWSD*).

1.4 SÍNTES DOS PROBLEMAS DE ROTEIRIZAÇÃO DE VEÍCULOS

No item 1.3 foram descritos os principais problemas de roteirização de veículos encontrados na literatura. Este item tem como objetivo apresentá-los de uma forma mais simplificada e genérica, a fim de facilitar o entendimento e mostrar a evolução e as diferenças fundamentais dos problemas.

A classificação completa dos problemas apresentados no item anterior está no quadro 1.1, que destaca as diferenças fundamentais de cada problema em relação ao problema clássico de roteirização de veículos. Conforme descrito no item 1.1, a função objetivo corresponde a um dos parâmetros mais relevantes dos problemas de roteirização de veículos, porém, não consta no quadro 1.1, pois pode variar para o mesmo tipo de problema. Em geral, os problemas de roteirização de veículos com janelas de tempo (VRPTW) buscam minimizar a distância total percorrida, além do tempo total de viagem, incluindo os tempos de espera. Já os problemas de roteirização com frota heterogênea (HFFVRP e FSMVRP) minimizam os custos fixos dos veículos e os custos variáveis em função da distância. Os demais problemas, em geral, têm como objetivo minimizar a distância total percorrida. Independente do tipo de problema, em alguns trabalhos busca-se minimizar também o número de veículos necessários.

Quadro 1.1 - Classificação dos principais problemas de roteirização de veículos

	Número Roteiros	Localização dos clientes	N. depósitos ou bases	Tipo de demanda	Restrições	N. Entregas por cliente	Variáveis de decisão	Período	Tipo de frota	Número de veículos
TSP	1	nós	1	determinística	—	1	roteiros de entrega	1 dia	1 veículo	limitado
CPP	1	arcos	1	determinística	—	1	roteiros de entrega	1 dia	1 veículo	limitado
MTSP	múltiplos	nós	1	determinística	—	1	roteiros de entrega	1 dia	homogênea	limitado
VRP	múltiplos	nós	1	determinística	capacidade do veículo	1	roteiros de entrega	1 dia	homogênea	limitado
MDVRP	múltiplos	nós	múltiplos	determinística	capacidade do veículo	1	roteiros de entrega	1 dia	homogênea	limitado
CARP	múltiplos	arcos	1	determinística	capacidade do veículo	1	roteiros de entrega	1 dia	homogênea	limitado
SVRP	múltiplos	nós	1	estocástica	capacidade do veículo	1	roteiros de entrega	1 dia	homogênea	limitado
VRPSD	múltiplos	nós	1	determinística	capacidade do veículo	>=1	roteiros de entrega qdade entregue a cada cliente	1 dia	homogênea	limitado
FSVRP	múltiplos	nós	1	determinística	capacidade do veículo	1	roteiros de entrega dimensionamento da frota	1 dia	homogênea	ilimitado
HFFVRP	múltiplos	nós	1	determinística	capacidade do veículo	1	roteiros de entrega	1 dia	heterogênea	limitado
FSMVRP	múltiplos	nós	1	determinística	capacidade do veículo	1	roteiros de entrega dimension. e compos. frota	1 dia	heterogênea	ilimitado
PVRP	múltiplos	nós	1	determinística	capacidade do veículo	1	roteiros de entrega	M dias	homogênea	limitado
TDVRP	múltiplos	nós	1	determinística	capacidade do veículo tempo dependente	1	roteiros de entrega	1 dia	homogênea	limitado
VRPTW	múltiplos	nós	1	determinística	capacidade do veículo janela de tempo	1	roteiros de entrega	1 dia	homogênea	limitado
VRPSTW	múltiplos	nós	1	determinística	capacidade do veículo janela de tempo flexível	1	roteiros de entrega	1 dia	homogênea	limitado
PDP Dial-a-ride	múltiplos	nós	1	determinística	capacidade do veículo precedência entre tarefas	1	roteiros de entrega	1 dia	homogênea	limitado

Legenda:

- TSP: Problema do Caixeiro Viajante
- CPP: Problema do Carteiro Chinês
- MTSP: Problema de Múltiplos Caixeiros Viajantes
- VRP: Problema Clássico de Roterização de Veículos
- MDVRP: VRP com múltiplos depósitos
- CARP: VRP com demanda em arcos
- SVRP: VRP com demanda estocástica
- VRPSD: VRP com entregas fracionadas
- FSVRP: VRP com dimensionamento de frota homogênea
- HFFVRP: VRP com frota heterogênea fixa
- FSMVRP: VRP com dimensionamento de frota heterogênea
- PVRP: VRP periódico
- TDVRP: VRP com tempo dependente
- VRPTW: VRP com janelas de tempo
- VRPSTW: VRP com janelas de tempo flexíveis
- PDP / dial-a-ride: Problema de coleta e entrega (VRP + precedência)

De acordo com o quadro 1.1, características como número de depósitos, tipo de demanda, restrições de janelas de tempo, etc, foram alteradas ou adicionadas ao VRP, resultando em variações e extensões do problema clássico de roteirização de veículos. Porém, as mesmas poderiam ser adicionadas ao TSP, CPP, MTSP ou a qualquer variação do VRP apresentada no quadro 1.1 (CARP, VRPSD, etc), resultando em novos problemas.

Obviamente, se fossem consideradas todas as combinações, o número de problemas seria muito grande. O objetivo deste capítulo foi identificar as características relevantes que determinam cada tipo de problema, para que seja tomada uma estratégia de solução adequada.

CAPÍTULO 2

ESTRATÉGIAS DE SOLUÇÃO EM PROBLEMAS DE ROTEIRIZAÇÃO DE VEÍCULOS: REVISÃO DA LITERATURA

Este capítulo apresenta, primeiramente, a complexidade dos problemas de roteirização de veículos, em especial dos problemas de roteirização de veículos com frota heterogênea, roteirização de veículos com janelas de tempo e roteirização com entregas fracionadas.

No item 2.2 são descritas as principais classificações das estratégias de solução em problemas de roteirização de veículos. Finalmente, no item 2.3 é feita uma revisão bibliográfica das principais estratégias de solução para cada uma das combinações do problema de roteirização de veículos estudado nesse trabalho.

2.1 COMPLEXIDADE COMPUTACIONAL DOS PROBLEMAS DE ROTEIRIZAÇÃO DE VEÍCULOS

A complexidade computacional pode ser definida como o comportamento de algoritmos para sua resolução e baseia-se no número de operações necessárias para resolver um dado problema. Quando, para um problema, existe um algoritmo polinomial para sua solução, o problema é chamado “tratável” ou de classe P (polinomial). Por outro lado, muitos problemas só podem ser resolvidos em tempos exponenciais e são ditos como “intratáveis” ou *NP*-completos (BODIN et al., 1983).

Os problemas de roteirização de veículos (VRP) variam quanto a sua complexidade, dependendo do número de variáveis e restrições que o problema considera em sua formulação e, mesmo com o avanço dos computadores, ainda há muitas dificuldades em sua resolução. 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. Devido ao seu caráter fortemente combinatório, a maioria

dos problemas práticos de roteirização de veículos é do tipo *NP*-completo (GOLDBARG; LUNA, 2000). 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. Para esses problemas complexos, a otimização utiliza técnicas para alcançar soluções próximas da ótima, como as heurísticas.

Lenstra e Rinnooy Kan (1981) analisaram a complexidade de problemas de roteirização de veículos, e concluíram que praticamente todos os problemas de roteirização de veículos são *NP*-completos (entre eles o problema do caixeiro viajante e o problema clássico de roteirização de veículos), pois não são resolvidos em tempo polinomial. De acordo com Bodin (1990), muitos dos problemas descritos na literatura modelam de forma simplificada os problemas reais.

Segundo Solomon e Desrosiers (1988), o problema de roteirização de veículos com janelas de tempo (VRPTW) também é *NP*-completo por ser uma extensão do VRP.

Gendreau et al. (1999), Dullaert et al. (2002) e Taillard (1999) afirmam que o problema de dimensionamento e roteirização de uma frota heterogênea de veículos (FSMVRP), o FSMVRP com janelas de tempo (FSMVRPTW) e o problema de roteirização com frota heterogênea fixa (HFFVRP), respectivamente, também são *NP*-completos por serem extensões do VRP.

Já o VRPSD, embora seja uma relaxação do VRP, também permanece *NP*-completo (DROR; TRUDEAU, 1990; ARCHETTI; MANSINI; SPERANZA, 2005).

Desta forma, o HFVRPTWSD que é uma combinação do problema de roteirização de veículos com frota heterogênea (HFVRP), do problema de roteirização com janelas de tempo (VRPTW) e do problema de roteirização com entregas fracionadas (VRPSD) também é *NP*-completo, justificando o uso de heurísticas e metaheurísticas para resolução do problema.

De acordo com Brejon (1998), há uma série de fatores que justificam o uso de heurísticas:

- os métodos exatos são proibitivos do ponto de vista computacional;

- os métodos heurísticos são mais simples, de fácil implementação e utilização e permite uma melhor compreensão das variáveis e dos parâmetros mais importantes do problema;
- os dados disponíveis são inexatos ou ilimitados, de modo que os erros cometidos na obtenção desses dados superam o erro causado pela não-otimalidade;
- um modelo mais realista com solução aproximada faz mais sentido que um modelo menos realista com solução ótima;
- necessidade de se resolver o problema continuamente, o que pode tornar vantajosa a utilização de um procedimento que tenha um menor custo computacional.

2.2 CLASSIFICAÇÃO DAS PRINCIPAIS ESTRATÉGIAS DE SOLUÇÃO EM PROBLEMAS DE ROTEIRIZAÇÃO E PROGRAMAÇÃO DE VEÍCULOS

2.2.1 CLASSIFICAÇÃO DAS ESTRATÉGIAS SEGUNDO BODIN E GOLDEN (1981) e BODIN ET AL. (1983)

Bodin e Golden (1981) e Bodin et al. (1983) classificaram as estratégias de solução para os problemas de roteirização de veículos da seguinte forma:

1) Agrupa – Roteiriza (*cluster first – route second*): consiste no procedimento de agrupar nós ou arcos de demanda primeiro, e depois construir rotas econômicas para cada agrupamento. Exemplos destas idéias são aplicadas nos trabalhos de Gillett e Miller (1974) e Fisher e Jaikumar (1981) para o problema básico de roteirização de veículos com um único depósito.

2) Roteiriza – Agrupa (*route first – cluster second*): primeiramente uma grande rota ou ciclo é construído (geralmente infactível), incluindo todas as entidades de demanda (nós e/ou arcos). Depois esta grande rota é dividida em um número menor e factível de rotas. Bodin e Berman (1979) utilizaram esta estratégia para roteirização de ônibus escolares.

3) Economias ou Inserções: procedimento de construção de uma solução em um dado caminho, de forma que, para cada etapa do processo, a configuração em construção, possivelmente infactível, é comparada com uma configuração alternativa que também deve ser infactível. A configuração alternativa é aquela que apresentar a maior economia em

termos de alguma função ou critério adotado, tal como custo total, ou a que inserir de forma menos custosa entidades de demanda ainda não inseridas na rota (ou rotas) em construção. O processo é concluído com uma configuração factível.

4) Melhoria/Troca: procedimento heurístico também conhecido como troca de arcos ou arestas onde em cada etapa uma solução factível é alterada, resultando em outra solução factível com o custo total reduzido. Procedimentos de troca são também conhecidos como procedimentos *k-opt*, onde *k* é o número de arcos ou arestas trocados a cada iteração.

5) Programação matemática: inclui algoritmos que são diretamente baseados em uma formulação matemática. Uma discussão mais profunda sobre este assunto pode ser encontrada em Magnanti (1981).

6) Otimização interativa: procedimento de propósito geral no qual há um alto grau de interação humana incorporada no processo de solução do problema. A idéia é que o tomador de decisões tem a experiência e a capacidade de colocar os parâmetros revisados e injetar correções subjetivas baseadas no conhecimento e na intuição do modelo de otimização.

7) Procedimento exato: Com exceção do problema do caixeiro viajante, os esforços no desenvolvimento de algoritmos exatos para problemas de roteirização e programação do tipo *NP*-completo têm sido muito limitados.

2.2.2 CLASSIFICAÇÃO SEGUNDO BOTT E BALLOU (1986)

Bott e Ballou (1986) classificaram os problemas de roteirização e programação de veículos em quatro categorias: heurísticas, algoritmos exatos, métodos iterativos e métodos combinados.

As heurísticas são divididas em quatro metodologias:

- Agrupa – Roteiriza (*cluster first – route second*)
- Roteiriza – Agrupa (*route first – cluster second*)
- Procedimentos de economias ou inserção
- Procedimentos de troca ou melhoria

Os métodos iterativos envolvem simulação e modelos de custeio. Os métodos combinados combinam pelo menos duas estratégias. Muitos estudos utilizam combinações de algoritmos exatos e heurísticas.

2.2.3 CLASSIFICAÇÃO DAS ESTRATÉGIAS DE SOLUÇÃO SEGUNDO SILVER, VIDAL E WERRA (1980), BALL E MAGAZINE (1981) E ZANAKIS, EVANS E VAZACOPOULOS (1989)

Silver, Vidal e Werra (1980), Ball e Magazine (1981) e Zanakis, Evans e Vazacopoulos (1989) classificam as heurísticas nas seguintes categorias: construção, melhoria, programação matemática, decomposição, partição, restrição do espaço de soluções e relaxação. Esta classificação é genérica, ou seja, não se restringe apenas a problemas de roteirização de veículos.

1. Construção: consiste em, passo a passo, adicionar componentes até a obtenção de uma solução factível. Os métodos construtivos ainda podem se diferenciar de acordo com a maneira como a rota é construída (seqüencial ou paralela) e o critério utilizado para expandir a rota.

2. Melhoria: inicia com uma solução factível e, através de uma seqüência de trocas ou outro tipo de movimento dentro de uma busca local, a solução é melhorada. Os métodos de melhoria são também conhecidos como métodos de busca em vizinhança, métodos de busca local ou métodos de descida.

3. Programação matemática: utiliza um modelo matemático e um procedimento exato de resolução e, através de uma modificação no procedimento exato, transforma-se em heurística.

4. Decomposição: consiste em resolver uma seqüência de problemas menores de tal forma que a saída de um seja a entrada para o próximo.

5. Partição: consiste em dividir o problema em subproblemas menores e independentes, de modo que a resolução de todos os subproblemas componha uma solução para o problema maior.

6. Restrição do Espaço de Soluções: o conjunto de soluções do problema é restrito de tal forma que o problema se torna mais fácil de resolver.

7. Relaxação: relaxando algumas restrições, pode-se obter facilmente uma solução factível para o problema original.

Segundo Zanakis, Evans e Vazacopoulos (1989) as heurísticas mais utilizadas são as construtivas e de melhoria.

No caso das heurísticas de melhoria, algumas dificuldades são encontradas no método de busca local, como sensibilidade à solução de partida, sensibilidade à vizinhança escolhida, sensibilidade à estratégia de escolha da próxima solução e possibilidade de um grande número de iterações. Para contornar algumas dificuldades da busca local podem-se considerar os seguintes aspectos:

- Redução da vizinhança: investigar um subconjunto da vizinhança da solução corrente, por exemplo, investigação aleatória;
- Multi-partida: repetir a busca local a partir de diferentes soluções;
- Multi-vizinhança: considera-se mais de uma vizinhança. Ao atingir um ótimo local com relação a uma vizinhança, inicia-se uma nova busca local baseada em outra vizinhança. O algoritmo termina quando a solução corrente é um ótimo local em relação a todas as vizinhanças empregadas;
- Segmentação da vizinhança: utilizada para aumentar a eficiência quando vizinhanças muito grandes são utilizadas. Pode ser vista como uma estratégia multi-vizinhança.

Cabe ressaltar que o desempenho dos métodos de busca local depende fortemente da definição da vizinhança. Para o problema do caixeiro viajante são utilizadas as seguintes vizinhanças: *city-swap* (troca de cidades ou clientes) e *k-opt* (troca de arestas).

2.2.4 CLASSIFICAÇÃO SEGUNDO LAPORTE (1992), CHRISTOFIDES (1985), OSMAN (1993) E CUNHA (1997)

Laporte (1992), Christofides (1985) e Osman (1993) classificaram os métodos de soluções em algoritmos exatos e heurísticas, enquanto Cunha (1997) propôs a classificação em três categorias: métodos exatos, métodos heurísticos e métodos emergentes.

Os métodos exatos garantem a solução ótima do problema. Como foi dito

anteriormente, a maioria dos problemas de roteirização de veículos pertencem à classe *NP*-completo. Os algoritmos de tempo polinomial para achar a solução ótima resolvem apenas problemas de pequeno porte, que não refletem a realidade. Por isso, pouca atenção tem sido dada à busca de soluções ótimas.

Os métodos heurísticos não garantem a solução ótima, mas geralmente resultam em soluções sub-ótimas de grande qualidade a um esforço computacional melhor.

Nicholson (1971) define heurísticas como um procedimento para resolver problemas através de um enfoque “intuitivo”, em geral racional, no qual a estrutura do problema possa ser interpretada e explorada inteligentemente para obter uma solução razoável.

Reeves (1993) define heurísticas como uma técnica que busca boas soluções (perto da ótima) com um custo operacional razoável, sem garantir soluções factíveis ou ótimas e, em muitos casos, não é capaz de afirmar quão próximo uma solução factível está da solução ótima. Além da teoria da complexidade computacional representar uma forte justificativa para a utilização de métodos heurísticos na solução de VRP, outro forte argumento apresentado pelo autor corresponde à possibilidade de modelar o problema real com maior precisão, uma vez que as heurísticas são mais flexíveis e aptas a operar com funções objetivos e/ou restrições mais complicadas e mais realistas do que os algoritmos exatos.

Os métodos emergentes reúnem técnicas mais recentes e avançadas, como as metaheurísticas busca tabu, algoritmos genéticos, *simulated annealing*, etc (CUNHA, 1997).

Segundo Souza (1993), *apud* Cunha (1997), embora os métodos emergentes possam ser classificados como métodos heurísticos, foram classificados isoladamente por utilizar dois ou mais procedimentos, com o objetivo de modificar e aprimorar gradualmente a solução.

O presente trabalho classificará os métodos de solução como algoritmos exatos, heurísticos e métodos emergentes ou metaheurísticas.

2.3 REVISÃO DA LITERATURA DOS PRINCIPAIS MÉTODOS DE SOLUÇÃO PARA PROBLEMAS DE ROTEIRIZAÇÃO DE VEÍCULOS

Este item faz uma revisão da literatura dos principais métodos de solução (métodos exatos, métodos heurísticos e metaheurísticas) para cada um dos problemas de roteirização de veículos relacionados com o problema estudado.

2.3.1 PROBLEMA CLÁSSICO DE ROTEIRIZAÇÃO DE VEÍCULOS (*VEHICLE ROUTING PROBLEM – VRP*)

a) MÉTODOS EXATOS

Christofides, Mingozzi e Toth (1981) implementaram algoritmos exatos para o problema clássico de roteirização de veículos com frota homogênea, baseados em relaxação lagrangiana e programação dinâmica relaxada dos problemas da árvore de cobertura mínima e do caminho mínimo. Além da restrição de capacidade dos veículos, há restrição de duração máxima da jornada de trabalho. O algoritmo final inclui reduções do problema e testes de dominância. O objetivo do modelo é determinar os roteiros de entrega de forma a minimizar a distância total percorrida. Os autores concluem que o algoritmo pode resolver de maneira ótima problemas com até 25 clientes.

Achuthan, Caccetta e Hill (2003) desenvolveram novos algoritmos de plano de corte, que foram implementados em um algoritmo *branch and cut*, para resolver um problema de roteirização de veículos. O problema considera frota homogênea com número fixo e variável de veículos e restrição de capacidade dos veículos. O objetivo do modelo é minimizar a distância total percorrida. O método foi aplicado para resolução de problemas de 15-100 clientes e os resultados foram comparados com outros métodos encontrados na literatura. Segundo os autores, o algoritmo obteve resultados significativos.

b) MÉTODOS HEURÍSTICOS

A heurística de economias de Clarke e Wright (1964) tem sido muito utilizada e tem apresentado bons resultados em problemas de roteirização de veículos. O método permite incorporar diversos tipos de restrição, como a inclusão de restrições de janelas de tempo.

Segundo Ballou (1993), a utilização deste algoritmo pode resultar em soluções próximas a 2% em relação à solução ótima.

O método de economias gera roteiros que respeitem as restrições de tempo (duração máxima da jornada de trabalho) e de capacidade, com o objetivo de minimizar a distância total percorrida pela frota. As restrições básicas do problema são:

- Cada rota inicia e termina no depósito;
- Cada cliente pertence somente a uma única rota;
- A demanda de cada cliente não pode exceder a capacidade do veículo;
- A demanda de todos os clientes de uma rota não pode exceder a capacidade do veículo;
- O tempo total de um roteiro não pode exceder a duração máxima da jornada de trabalho do motorista.

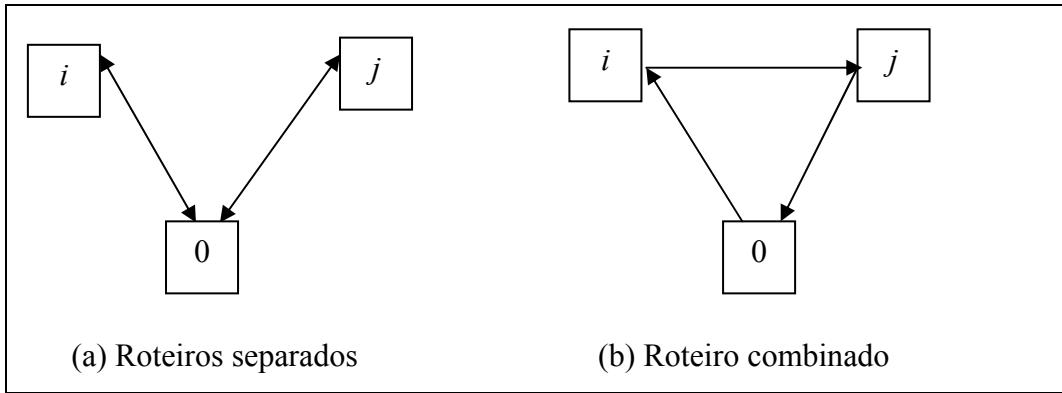
O algoritmo de Clarke e Wright constitui um modelo heurístico do tipo *saving* (economia). O método de Clarke e Wright inicia com a pior situação, em que cada veículo sai do CD para atender um único cliente e retorna ao depósito. Suponhamos que dois clientes (i e j) são atendidos, cada um por um único veículo. Considerando $d_{0,i}$ e $d_{0,j}$ as distâncias entre o depósito (0) e os clientes i e j , a distância total percorrida pode ser definida como:

$$D = 2 \times (d_{0,i} + d_{0,j}) \quad (2.1)$$

Uma forma de minimizar a distância total D seria juntar os dois clientes i e j em um único roteiro. A distância total percorrida D' passaria a ser:

$$D' = d_{0,i} + d_{i,j} + d_{0,j} \quad (2.2)$$

Figura 2.1 - Cominando dois clientes em um roteiro



Logo, a economia gerada pela junção dos clientes em um único roteiro seria:

$$s_{ij} = D - D' = d_{0,i} + d_{0,j} - d_{i,j} \quad (2.3)$$

A seguir é descrito o algoritmo completo de Clarke e Wright.

Figura 2.2 - Passos iniciais do algoritmo de economias

Passo 1: Combinar todos os pontos (clientes) dois a dois e calcular a economia para cada combinação, de acordo com a fórmula acima.

Passo 2: Ordenar as combinações em ordem decrescente de economia.

Passo 3: Começar do topo da lista, com a combinação que gerou maior ganho.

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

1. Versão Paralela: efetua a melhor união factível;
2. Versão Seqüencial: realiza a extensão máxima de uma rota.

Figura 2.3 - Versão paralela do algoritmo de economias

Passo 4: Se, ligando os pares de nós i e j , o resultado for uma rota factível de acordo com as restrições do problema, fazer a união; caso contrário, elimine-a.

Passo 5: Se ainda houver economia, tentar a união com o próximo par de nós da lista e volte ao Passo 4. Senão houver mais economias, o algoritmo termina.

Figura 2.4 - Versão seqüencial do algoritmo de economias

Passo 4: Se, ligando os pares de nós i e j , resultar uma rota factível, de acordo com as restrições do problema de roteirização de veículos, fazer esta união e ir para o passo 6. Senão, vá para o passo 5.

Passo 5: Se a rota não puder mais ser estendida, resultando em uma rota infactível, terminar a rota e iniciar uma nova rota com o par de nós do passo 4.

Passo 6: Repetir os passos 4 e 5 enquanto houver alguma economia na lista.

A versão paralela da heurística de Clarke e Wright, por utilizar sempre os arcos que proporcionam as maiores economias, gera, na grande maioria das vezes, soluções de menor custo que a versão seqüencial. Isto se deve ao fato de a versão seqüencial ser míope, trabalhando uma rota por vez. Por outro lado, esta característica da versão seqüencial faz com que ela priorize o aproveitamento de cada rota, de modo que a solução final gere, na maioria das vezes, menos rotas do que a versão paralela. A eficiência da versão adotada varia de acordo com o tipo de problema (CAMPOS, 2004).

A heurística de Clarke e Wright (versão paralela) será adaptada para o problema estudado neste trabalho, com algumas modificações, sendo utilizada como uma das soluções iniciais para a metaheurística *scatter search*.

Muitas modificações têm sido propostas para a heurística de Clarke e Wright (1964). Outras heurísticas baseadas no critério de economias, além daquelas apresentadas neste trabalho, podem ser encontradas em Mole e Jameson (1976), Paessens (1988) e Landeghem (1988).

Gillet e Miller (1974) implementaram o método de duas-fases (agrupa-roteiriza) para um problema de roteirização de veículos com frota homogênea e um total de 250 nós. A primeira fase utiliza a heurística da varredura para agrupar os clientes aos veículos e na segunda fase calcula-se o custo mínimo do problema do caixeiro viajante para cada uma das rotas. O objetivo do modelo é determinar os roteiros de entrega que minimizam a distância total percorrida, de forma que as restrições de capacidade de veículo e distância máxima (duração máxima da jornada de trabalho) de cada veículo sejam respeitadas.

Fisher e Jaikumar (1981) analisaram o problema de roteirização de veículos com frota homogênea, e também utilizaram o método agrupa-roteiriza. Este modelo também pode vir acompanhado de técnicas de relaxações quando o problema a ser tratado for mais complexo. O método foi aplicado em 12 problemas encontrados na literatura e comparado com as seguintes heurísticas: Clarke e Wright (1964), Gillet e Miller (1974) e 2 heurísticas de Christofides, Mingozi e Toth (1978). Segundo os autores, o método superou as demais heurísticas em termos de custo total de viagem e tempo computacional, com exceção do tempo computacional obtido no algoritmo de Clarke e Wright.

Desrochers e Verhoog (1989) apresentaram uma extensão da heurística de economias para o problema clássico de roteirização de veículos. As economias são baseadas em sucessivas fusões de rotas. A cada iteração, a melhor fusão é selecionada através do algoritmo MBSA (*Matching Based Saving Algorithm*). Segundo os autores, este critério pode ser considerado menos míope do que as heurísticas de construção usuais. O objetivo é determinar um conjunto de roteiros de forma a minimizar os custos de roteirização, respeitando as restrições de capacidade dos veículos e duração máxima da jornada de trabalho.

c) BUSCA TABU

Taillard (1993) implementou busca tabu como um método de busca iterativa para um problema de roteirização de veículos. Dois métodos de partição foram implementados. O primeiro, baseado em regiões polares, é apropriado para problemas euclidianos cujas cidades são regularmente distribuídas ao redor do depósito central. Já o segundo, apresentado por Taillard, pode ser aplicado a qualquer problema. O problema tem como característica frota de veículos homogênea e o objetivo é determinar os roteiros para cada veículo de forma que a distância total percorrida pelos veículos seja minimizada, a demanda de cada cidade seja atendida e as restrições de capacidade dos veículos e duração máxima da jornada de trabalho sejam respeitadas.

Osman (1993) desenvolveu uma metaheurística híbrida (*simulated annealing* e busca tabu) para um problema clássico de roteirização de veículos. Osman implementou, com sucesso, procedimentos de melhoria baseado em trocas entre rotas (λ -*interchange*).

Gendreau, Hertz e Laporte (1994) implementaram um procedimento busca tabu

sofisticado para um problema de roteirização de veículos com frota homogênea e restrições de capacidade dos veículos e duração máxima da jornada de trabalho. O algoritmo utilizado é uma adaptação do procedimento de inserção generalizada (GENI – *Generalized Insertion Procedure*) desenvolvido por Gendreau et al. (1992) e engloba heurísticas construtivas e de melhoria. O procedimento permite que as soluções se tornem temporariamente infactíveis através de penalidades. O objetivo do modelo é minimizar a distância total percorrida.

Barbarosoglu e Ozgur (1999) desenvolveram um algoritmo busca tabu baseado nas idéias de Taillard (1993), para um problema de roteirização de veículos (VRP). A frota de veículos é homogênea e há restrições de capacidade dos veículos e duração máxima da jornada de trabalho. Os autores propõem um novo procedimento de busca de vizinhança sem qualquer diversificação e um novo esquema de intensificação. Segundo os autores, os resultados numéricos mostraram que o desempenho do algoritmo é semelhante aos melhores resultados encontrados na literatura.

Kelly e Xu (1999) desenvolveram uma metaheurística busca tabu para o problema clássico de roteirização de veículos. O objetivo é minimizar a distância total percorrida, garantindo o atendimento de todos os clientes e a não-violão da restrição de capacidade dos veículos. O algoritmo consiste em duas etapas. Na primeira utilizou-se uma heurística construtiva de forma a gerar rotas diversas. Na segunda foi implementada uma heurística de melhoria baseada em particionamento de conjuntos e a metaheurística busca tabu, que combinou as rotas obtidas com técnicas de busca local baseadas em Taillard (1993), criando novas rotas. O desempenho do algoritmo foi comparado com o conjunto de problemas proposto por Christofides et al. (1979) e, segundo os autores, apresentou ótimos resultados.

d) ALGORITMOS GENÉTICOS

Baker e Ayechew (2003) implementaram um algoritmo genético para um problema de roteirização de veículos (VRP) com frota homogênea e restrições de capacidade dos veículos e duração máxima da jornada de trabalho. A população inicial foi obtida com base em dois métodos: o primeiro é baseado no trabalho de Gillet e Miller (1974) e o segundo, no trabalho de Fischer e Jaikumar (1981). O processo reprodutivo foi baseado na escolha aleatória de dois indivíduos da população. O *crossover* foi aplicado em 2 pontos para um problema com 20 clientes. Para acelerar a convergência foram implementadas melhorias. Os resultados foram

comparados com os trabalhos de Osman (1993), Taillard (1993), Rochat e Taillard (1995), Gendreau, Hertz e Laporte (1994). Segundo os autores, este modelo é competitivo com outras metaheurísticas para VRP, como busca tabu e *simulated annealing*, tanto em termos de tempo computacional quanto de qualidade da solução.

2.3.2 PROBLEMA DE ROTEIRIZAÇÃO DE VEÍCULOS COM JANELAS DE TEMPO (VEHICLE ROUTING PROBLEM WITH TIME WINDOWS – VRPTW)

a) MÉTODOS EXATOS

Baker (1982) utilizou a estratégia *branch and bound* com algoritmo de etiquetamento para um problema de roteirização com um único veículo e restrições de janelas de tempo e capacidade do veículo. O objetivo do modelo é minimizar o tempo total das rotas.

Kolen et al. (1987) utilizaram um algoritmo *branch and bound* para resolução de um problema de roteirização de veículos com restrições de janela de tempo (VRPTW). Os clientes são atendidos por uma frota de veículos homogênea. O objetivo é determinar um conjunto de rotas factíveis, de forma a minimizar a distância total percorrida, respeitando as restrições de janela de tempo e capacidade dos veículos.

Desrochers, Desrosiers e Solomon (1992) implementaram o método de geração de colunas para solução do problema de roteirização e programação de veículos com janelas de tempo (VRPTW) e frota homogênea. Os autores assumem que o número de veículos é ilimitado, o que faz com que o tamanho da frota seja determinado simultaneamente com os roteiros (FSVRPTW). O objetivo do modelo é minimizar a distância total percorrida. Os autores aplicam o método de solução ao conjunto de problemas de propostos por Solomon (1987), resolvendo os problemas dos conjuntos R1, C1 e RC1. Na maior parte dos problemas com 100 clientes, não foi possível obter solução ótima. Em função disto, os autores modificaram os problemas originais, criando, a partir de cada problema, dois outros, com 50 e 25 clientes. Em todos os problemas com 25 clientes obteve-se a solução ótima. De qualquer maneira, conseguiu-se resolver alguns problemas originais. O maior problema resolvido envolvia 100 clientes e 18 veículos. Comparando-se com o maior problema resolvido por métodos exatos até então, esse trabalho representou uma significativa contribuição na

pesquisa de métodos exatos de resolução de problemas de roteirização e programação com janela de tempo.

Bard, Kontoravdis e Yu (2002) desenvolveram um procedimento *branch and cut* para o VRPTW com frota homogênea. O objetivo é minimizar o número de veículos necessários e a distância total percorrida, garantindo que as restrições de capacidade dos veículos e de janelas de tempo sejam respeitadas. Além dos roteiros, busca-se determinar o número de veículos necessários (FSVRPTW). O método proposto foi testado em seis conjuntos de problemas (R1, C1, RC1, R2, C2, RC2) de Solomon (1987).

b) MÉTODOS HEURÍSTICOS

Solomon (1986) implementou heurísticas para o problema de roteirização e programação de veículos com restrição de janela de tempo, analisando o comportamento do pior-caso (*worst-case performance*). A frota de veículos é homogênea e há restrições de capacidade dos veículos e janelas de tempo. O objetivo é minimizar o número de veículos necessários, a distância total percorrida e o tempo total das rotas. Foram implementadas diversas heurísticas construtivas (economias, vizinho mais próximo, inserção, método agrupa-roteiriza, roteiro gigante) e heurística de melhoria baseada na troca de arcos.

Solomon (1987) desenvolveu sete heurísticas construtivas para o problema de roteirização de veículos com janela de tempo. A frota de veículos é homogênea e há restrições de capacidade dos veículos e janelas de tempo. O autor assume que o número de veículos é ilimitado, o que faz com que o tamanho da frota seja determinado simultaneamente com os roteiros (FSVRPTW). Segundo Solomon, para uma heurística ser eficiente na solução de um problema de roteirização com restrição temporal, é fundamental a forma como as restrições são incorporadas no método. O objetivo do modelo é minimizar a distância total percorrida, além do tempo total das rotas, que inclui os tempos de espera. Os principais parâmetros utilizados são:

e_i – início da janela de tempo do cliente i

l_i – fim da janela de tempo do cliente i

b_i – instante de início efetivo do serviço no cliente i

s_i – tempo de duração do serviço no cliente i

t_{ij} – tempo de viagem do cliente i ao cliente j

d_{ij} – distância do cliente i ao cliente j

w_{ij} – tempo de espera no cliente i

c_{ij} – custo de deslocamento do cliente i ao cliente j , onde:

$$c_{ij} = \rho_1 d_{ij} + \rho_2 (b_j - b_i) \quad (2.4)$$

O início do atendimento do cliente j é o horário máximo entre o instante de chegada no cliente j e o início da janela de tempo (e_i):

$$b_j = \max \{e_j, b_i + s_i + t_{ij}\} \quad (2.5)$$

A seguir são descritas as sete heurísticas propostas por Solomon.

1. Heurística de economia

É uma extensão da heurística original proposta por Clarke e Wright (1964), adicionando restrições de janela de tempo. Primeiramente, passa-se a considerar a orientação da rota devida à incorporação da restrição de janela de tempo. Em segundo lugar é necessário verificar a viabilidade temporal para união de duas rotas, de forma que as restrições de janelas de tempo sejam respeitadas.

Esta heurística começa com n rotas distintas, sendo que cada cliente i ($i = 1, \dots, n$) é servido por um único veículo (frota dedicada). O critério de economia utilizado é o mesmo da heurística original de Clarke e Wright, buscando maximizar a função economia em função da distância.

2. Heurística de economia com limite de tempo de espera

Para levar em consideração características de espaço e tempo, adiciona-se um limite de tempo de espera. Portanto, essa heurística é semelhante à heurística de economia, acrescentando um critério de limite de tempo de espera na união de duas rotas. Esta heurística evita a união de clientes distantes temporalmente.

3. Heurística do vizinho mais próximo com orientação temporal

Essa heurística pertence à classe seqüencial. Inicia-se uma rota com o cliente mais próximo ao depósito. O critério de proximidade é calculado em função de aspectos geográficos, temporais e urgência no atendimento. A cada iteração adiciona-se o cliente mais próximo, até que não seja viável mais nenhuma inserção nessa rota. Rotas são iniciadas enquanto houver cliente a ser incluso. A medida de proximidade entre os clientes i e j é C_{ij} , sendo que:

$$C_{ij} = \delta_1 d_{ij} + \delta_2 T_{ij} + \delta_3 v_{ij}, \quad \begin{aligned} \delta_1 + \delta_2 + \delta_3 &= 1 \\ \delta_1 \geq 0, \delta_2 \geq 0, \delta_3 &\geq 0 \end{aligned} \quad (2.6)$$

onde:

d_{ij} = distância entre os clientes i e j

T_{ij} = intervalo de tempo entre o final do serviço i e o início do serviço j

$$T_{ij} = b_j - (b_i + s_i) \quad (2.7)$$

v_{ij} = urgência do cliente j , corresponde ao tempo disponível entre o final da janela de tempo do cliente j e a chegada do veículo em j

$$v_{ij} = l_j - (b_i + s_i + t_{ij}) \quad (2.8)$$

A cada iteração, esta heurística insere o cliente j com menor valor de C_{ij} , ou seja, busca-se o cliente mais próximo espacialmente, temporalmente e com maior urgência de atendimento.

4. Heurística de inserção II, com critério que minimiza o acréscimo de tempo e distância causados pela inserção de um cliente

Esta heurística pertence à classe seqüencial. O primeiro passo é o início de uma rota com um determinado cliente. Os critérios de início das rotas são:

- a) cliente mais distante ainda não alocado
- b) cliente com menor valor de l_i (fim da janela de tempo do cliente i)

Os critérios de início são exclusivos, ou seja, utiliza-se um ou outro. Iniciada uma rota com um determinado cliente, o próximo passo é a utilização de um critério para inserir o próximo cliente da rota. Considera-se (i_0, i_1, \dots, i_m) a rota atual com $i_0 = i_m = 0$ (depósito) e $m-1$ clientes. O método de inserção utiliza dois critérios, $c_1(i, u, j)$ e $c_2(i, u, j)$, a cada iteração, para inserir um novo cliente u , entre os clientes i e j da rota.

O primeiro critério tem como objetivo determinar a melhor posição de inserção de um determinado cliente u . Calcula-se para cada cliente u ainda não alocado a nenhuma rota, a melhor posição p viável de inserção, definida por $c_1(i(u), u, j(u))$:

$$c_1(i(u), u, j(u)) = \text{melhor posição de inserção do cliente } u$$

$$c_1(i(u), u, j(u)) = \min \left\{ c_1(i_{p-1}, u, i_p) \right\} \quad p = 1, \dots, m \quad (2.9)$$

onde:

$$c_1(i, u, j) = \alpha_1 c_{11}(i, u, j) + \alpha_2 c_{12}(i, u, j), \quad \alpha_1 + \alpha_2 = 1$$

$$\alpha_1, \alpha_2 \geq 0 \quad (2.10)$$

$$c_{11}(i, u, j) = d_{iu} + d_{uj} - u d_{ij}, \quad u \geq 0$$

$$c_{12}(i, u, j) = b_{j/u} - b_j \quad (2.11)$$

onde $b_{j/u}$ é o novo tempo que inicia o serviço do cliente j após a inserção do cliente u na rota.

A função c_{11} corresponde ao acréscimo de distância após a inserção do cliente u . A função c_{12} é a diferença entre o novo instante de início de serviço do cliente j e o instante anterior, em função da inserção do cliente u . A função c_1 , portanto, corresponde ao acréscimo de distância e tempo causado pela inserção do cliente u . A melhor posição de inserção de um cliente é aquela que minimiza esse acréscimo.

O segundo critério tem como objetivo determinar o cliente u a ser inserido na rota entre os clientes i e j , que é aquele que maximiza a função $c_2(i(u), u, j(u))$:

$$\begin{aligned} c_2(i(u^*), u^*, j(u^*)) &= \max [c_2(i(u), u, j(u))] \\ \text{e, } c_2(i, u, j) &= \lambda d_{0u} - c_1(i, u, j), \quad \lambda \geq 0. \end{aligned} \quad (2.12)$$

A função c_2 é a diferença (utilizando um critério de ponderação λ) entre a distância direta da base ao cliente u e o acréscimo em tempo e distância causados pela inserção da função c_1 . Por exemplo, para dois clientes com mesmo valor para a função c_1 , o mais distante da base seria inserido.

5. Heurística de inserção I2, com critério que minimiza o tempo e a distância total da rota

Esta heurística é semelhante à anterior, com algumas diferenças. A primeira diferença é que se acrescenta ao critério de início cliente não alocado com a menor soma de tempo e distância ao depósito. Os critérios de início das rotas podem ser, portanto:

- a) cliente mais distante ainda não alocado
- b) cliente com menor valor de l_i (fim da janela de tempo do cliente i)
- c) cliente com a menor soma de tempo e distância ao depósito

O primeiro critério de inserção $c_1(i, u, j)$ para escolha da melhor posição de cada cliente é semelhante ao anterior. O segundo critério tem como objetivo determinar o cliente u a ser inserido na rota entre os clientes i e j , que é aquele que minimiza a função $c_2(i(u), u, j(u))$:

$$\begin{aligned}
c_2(i(u^*), u^*, j(u^*)) &= \min [c_2(i(u), u, j(u))], \text{ e} \\
c_2(i, u, j) &= \beta_1 R_d(u) + \beta_2 R_t(u), \quad \beta_1 + \beta_2 = 1 \\
\beta_1, \beta_2 &\geq 0
\end{aligned} \tag{2.13}$$

onde:

R_d e R_t são respectivamente a distância e a duração total da rota sendo construída, considerando a inclusão do cliente u .

Portanto, essa heurística, ao construir uma rota, busca selecionar clientes cujos custos de inserção minimizam uma ponderação da distância e da duração total da rota que está sendo construída.

6. Heurística de inserção I3

Esta heurística é semelhante à I1, com algumas diferenças. A primeira diferença é que se acrescenta ao critério de início cliente não alocado com a menor soma de tempo e distância ao depósito, da mesma forma que em I2. Os critérios de início das rotas podem ser, portanto:

- a) cliente mais distante ainda não alocado
- b) cliente com menor valor de l_i (fim da janela de tempo do cliente i)
- c) cliente não alocado com a menor soma de tempo e distância ao depósito

O primeiro critério de inserção $c_1(i, u, j)$ para escolha da melhor posição de cada cliente passa a ser:

$$\begin{aligned}
c_1(i, u, j) &= \alpha_1 c_{11}(i, u, j) + \alpha_2 c_{12}(i, u, j) + \alpha_3 c_{13}(i, u, j), \quad \alpha_1 + \alpha_2 + \alpha_3 = 1 \\
\alpha_1, \alpha_2, \alpha_3 &\geq 0
\end{aligned} \tag{2.14}$$

onde:

$c_{11}(i, u, j)$ e $c_{12}(i, u, j)$ não se alteram em relação à I1,

$$c_{13}(i, u, j) = l_u - b_u$$

O segundo critério tem como objetivo determinar o cliente u a ser inserido na rota entre os clientes i e j , que é aquele que maximiza a função $c_2(i(u), u, j(u))$:

$$c_2(i(u^*), u^*, j(u^*)) = \max [c_2(i(u), u, j(u))], \text{ onde:} \quad (2.15)$$

$$c_2(i, u, j) = c_1(i, u, j)$$

Portanto, essa heurística acrescenta a urgência do cliente, de acordo com a função c_{13} , ao aspecto temporal do primeiro critério de inserção.

7. Heurística da varredura com orientação temporal

Esta heurística decompõe o problema principal em dois subproblemas: um de agrupamento e outro de roteirização/programação (método agrupa-roteiriza). O primeiro problema agrupa os clientes em rotas, e cada rota é servida por um determinado veículo. No segundo subproblema resolve-se um problema de roteirização para cada agrupamento definido na fase anterior. O primeiro subproblema é resolvido aplicando-se a primeira fase da heurística original de varredura de Gillet e Miller (1974). Já para o segundo subproblema de roteirização de cada agrupamento, aplica-se a heurística de inserção II.

Análise dos resultados das sete heurísticas

Solomon aplicou as heurísticas propostas em seis conjuntos de problemas, em um total de 56 problemas, conforme mostra o quadro abaixo.

Quadro 2.1 - Conjunto de problemas de Solomon (1987)

Conjunto	Problemas
R1	R101-R112
R2	R201-R211
C1	C101-C109
C2	C201-C208
RC1	RC101-RC108
RC2	RC201-RC208

A distribuição geográfica dos clientes segue as seguintes regras:

- nos conjuntos R1 e R2 a posição dos clientes é gerada randomicamente por uma distribuição uniforme.
- nos conjuntos C1 e C2 os clientes estão em grupos (*clusters*)
- nos conjuntos RC1 e RC2, os clientes estão em semi-agrupamentos (*semi-clusters*), ou seja, parte dos clientes estão localizados randomicamente e parte dos clientes estão localizados em grupos.

Cada problema é formado por 100 clientes. A porcentagem de clientes com janela de tempo é de 100, 75, 50 ou 25%, conforme o problema. O horizonte de planejamento é curto para os problemas dos conjuntos R1, C1, RC1 e longo para os problemas dos conjuntos R2, C2 e RC2. Para cada problema, a distribuição geográfica dos clientes, a demanda e o tempo de serviço não se alteram; altera-se apenas a porcentagem de clientes com janela de tempo. Solomon procura identificar o efeito de alguma característica particular sobre o comportamento das diversas heurísticas.

Aplicando cada procedimento ao conjunto de problemas, Solomon concluiu que a heurística de inserção I1 apresentou melhores resultados. Segundo o autor, para problemas com restrições de janela de tempo, o critério de inserção prioriza esses aspectos temporais. O segundo melhor resultado foi obtido pela heurística I2.

Cabe lembrar que uma das heurísticas construtivas implementadas neste trabalho é uma adaptação da heurística de inserção I1 de Solomon.

Solomon, Baker e Schaffer (1988) implementaram heurísticas de melhoria em um problema de roteirização e programação de veículos com janelas de tempo, baseado nas idéias de Psaraftis (1983). O desempenho das heurísticas é medido através da distância total percorrida e do tempo total das rotas, incluindo os tempos de espera. O problema tem como característica frota homogênea e restrições de capacidade de veículo e janelas de tempo. Foram desenvolvidos métodos para acelerar os procedimentos de melhoria por troca de arcos, com o objetivo de reduzir o grande tempo computacional das heurísticas de troca de arcos *k-opt*, através da eliminação de testes de viabilidade desnecessários. As heurísticas foram

aplicadas ao conjunto de problemas proposto por Solomon (1987) com restrição de janela de tempo. Também foram implementados procedimentos de melhoria já existentes (*2-opt* e *3-opt*) para o conjunto de problemas de Solomon (1987) sem janelas de tempo. Os autores concluem que a inclusão de janelas de tempo não gerou maior complexidade computacional, pois os métodos de aceleração para eliminar testes de viabilidade desnecessários obtiveram uma boa performance. O trabalho de Thompson e Psaraftis (1993) apresenta um procedimento de melhoria que obteve resultados superiores.

Thompson e Psaraftis (1993) propõem um procedimento de melhoria baseada na transferência cíclica (*cycle transfer*) para resolução de problemas de roteirização e programação de veículos, entre eles o VRPTW e o VRP, ambos com frota homogênea. No VRP, o objetivo é minimizar a distância total percorrida. No VRPTW há três objetivos que estão ordenados de forma hierárquica. O primeiro objetivo é minimizar o número de veículos necessários. O segundo objetivo é minimizar o tempo total das rotas, que consiste na soma dos tempos de viagem, tempo de espera e tempo de serviço. O terceiro e último objetivo é minimizar a distância total percorrida.

O procedimento de transferência cíclica pode ser classificado como um procedimento de melhoria, que envolve troca de arcos entre as rotas, mas, em princípio, é auto-suficiente para gerar soluções por si próprio (BREJON, 1998). No caso do VRPTW, o procedimento foi aplicado ao conjunto de problemas proposto por Solomon (1987) e, segundo os autores, se mostrou promissor, pois obteve resultados superiores às melhores soluções obtidas por Solomon, Baker e Schaffer (1988).

Potvin e Rousseau (1993) propõem uma adaptação da heurística de inserção seqüencial I1 de Solomon (1987), para uma frota de veículos homogênea, através de um método de construção paralela, baseado nas idéias de Altinkemer e Gavish (1991), no qual as rotas são construídas simultaneamente, tanto no critério de início de rotas quanto de inserção. Segundo os autores, a heurística I1 faz com que os últimos clientes a serem alocados fiquem geograficamente espalhados. Quando isso acontece, as últimas rotas construídas geralmente são de pior qualidade, devido à miopia típica dos métodos seqüenciais. O algoritmo minimiza a distância total percorrida (ou tempo total de percurso), além do tempo de espera. Os autores concluem que o algoritmo de construção paralela se mostrou melhor que a heurística de inserção I1 de Solomon no conjunto de problemas RC, ligeiramente melhor no conjunto de

problemas R, e pior no conjunto de problemas C. Conclui-se, portanto, que o método de construção paralela é indicado para problemas reais que não são caracterizados por um grande agrupamento dos clientes, caso em que a heurística de Solomon se mostra mais adequada.

Russell (1995) implementou heurísticas construtivas (inserção paralela) e de melhoria (troca de nós entre rotas) em um problema de roteirização de veículos com janelas de tempo (VRPTW) e frota homogênea. Há restrições de capacidade dos veículos, janelas de tempo e duração máxima da jornada de trabalho. Primeiramente busca-se minimizar o número de veículos necessários, em segundo lugar minimiza-se o tempo total das rotas e, finalmente, a distância total percorrida. As heurísticas foram testadas em 6 conjuntos de problemas de Solomon (1987) e em 2 problemas reais de roteirização.

Potvin e Rousseau (1995) implementaram heurísticas de melhoria (*2-opt* e troca de nós) para problemas de roteirização de veículos com restrições de janelas de tempo, com e sem restrição de capacidade dos veículos. A frota de veículos é homogênea. Primeiramente busca-se minimizar o número de veículos necessários e, em segundo lugar, o tempo total das rotas (soma dos tempos de viagem, tempo de espera e tempo de serviço). As soluções iniciais foram geradas pela heurística de inserção I1 de Solomon (1987). As heurísticas foram aplicadas ao conjunto de problemas de Solomon e a um conjunto de problemas euclidianos aleatórios gerados pelos autores.

Cunha (1997) implementou heurísticas baseadas na relaxação lagrangiana para um problema de roteirização de veículos com restrições de janela de tempo, capacidade dos veículos e duração máxima da jornada de trabalho. São relaxadas as restrições relacionadas ao atendimento de todos os clientes uma única vez. Como o problema relaxado ainda é de difícil solução, utiliza-se um algoritmo de etiquetamento permanente para o problema do caminho mínimo com janelas de tempo. Três diferentes heurísticas foram desenvolvidas a partir desta estratégia de solução. Duas delas são aplicadas apenas em problemas com frota homogênea, enquanto a terceira (heurística de agrupamento e alocação seqüencial) pode ser aplicada em problemas com frota heterogênea.

As heurísticas foram aplicadas ao conjunto de problemas proposto por Solomon (1987). A heurística de agrupamento e alocação seqüencial apresentou resultados superiores aos de Solomon (1987) e as outras duas heurísticas tiveram bons resultados em alguns

conjuntos de problemas. Com base nestes resultados, aplicou-se a heurística de agrupamento e alocação seqüencial em um problema real de distribuição com um total de 136 entregas. Com a implementação das heurísticas foi possível reduzir a distância total percorrida, os custos operacionais e o número de veículos necessário.

c) BUSCA TABU

Garcia, Potvin e Rousseau (1994) descrevem uma implementação paralela da metaheurística busca tabu para um problema de roteirização e programação de veículos com janela de tempo (VRPTW). O objetivo é atender as demandas dos clientes ao menor custo possível, respeitando as restrições de capacidade dos veículos e janelas de tempo. O custo de roteirização inclui não só a distância total percorrida (tempo total de viagem), mas também o tempo total de espera. A solução inicial foi gerada pela heurística de inserção de Solomon e os resultados finais foram comparados com o conjunto de problemas proposto por Solomon (1987).

Potvin et al. (1996) implementaram busca tabu para um problema de roteirização de veículos com janelas de tempo. Dado um depósito central, uma frota de veículos homogênea e um conjunto de clientes com demandas conhecidas, o objetivo é determinar um conjunto de rotas, originando e terminando no depósito, que atenda todos os clientes ao menor custo possível. O objetivo é minimizar, primeiramente, o número de veículos necessários e, em segundo lugar, a distância total percorrida (ou tempo viagem), além do tempo de espera, respeitando as restrições de capacidade dos veículos e janelas de tempo. A solução inicial foi gerada com base na heurística de inserção I1 de Solomon (1987).

Badeau et al. (1997) implementaram um algoritmo busca tabu paralelo para um problema de roteirização de veículos com janelas de tempo (VRPTW). Os autores consideram janelas de tempo rígidas e flexíveis (VRPSTW). A frota de veículos é homogênea e há restrições de janelas de tempo e capacidade dos veículos. O objetivo é minimizar a distância total percorrida, respeitando as restrições do problema. A vizinhança é baseada em movimentos de troca de arcos (*2-opt*), troca cruzada e trocas de clientes na mesma rota.

Chiang e Russell (1997) desenvolveram uma metaheurística busca tabu reativa para um problema de roteirização de veículos com janelas de tempo (VRPTW). A busca tabu

reativa varia dinamicamente o tamanho e os movimentos da lista tabu, evitando que a busca estacione em um ótimo local. O problema tem como característica frota homogênea e restrições de capacidade do veículo, janelas de tempo e duração máxima da jornada de trabalho. O objetivo é minimizar o número de veículos necessários, o tempo total das rotas e a distância total percorrida. A solução inicial é similar à heurística de inserção de Solomon (1987), porém as rotas são construídas de forma paralela. As heurísticas foram testadas no conjunto de problemas de Solomon (1987) e em 2 problemas reais de Russell (1995).

Tan et al. (2001) implementaram diversas metaheurísticas (busca tabu, *simulated annealing* e algoritmos genéticos) para um problema de roteirização de veículos com janelas de tempo e frota homogênea. O objetivo é minimizar a distância total percorrida, garantindo que as restrições de capacidade, janelas de tempo e duração máxima da jornada sejam respeitadas. A solução inicial foi obtida pela heurística de inserção de Solomon (1987). As metaheurísticas foram aplicadas aos 56 problemas de Solomon e, segundo os autores, superaram ou igualaram 18 das melhores soluções encontradas na literatura.

d) GRASP

Kontoravdis e Bard (1995) implementaram o procedimento GRASP – *Greedy Randomized Adaptive Search Procedure* (Procedimento de Busca Adaptativa, Aleatória e Gulosa) para um problema de roteirização de veículos com janelas de tempo (VRPTW). O primeiro objetivo é minimizar o número de veículos necessários. Em segundo lugar, busca-se minimizar a distância total percorrida. O algoritmo foi testado no conjunto de problemas de Solomon (1987), além de problemas reais com até 417 clientes.

e) ALGORITMOS GENÉTICOS

Potvin e Bengio (1996) propuseram um algoritmo genético chamado *GENEROUS (GENEtic ROUTing System)* para um problema de roteirização de veículos com janela de tempo (VRPTW). Dado um depósito central, uma frota de veículos homogênea e um conjunto de clientes com demandas conhecidas, o problema consiste em determinar um conjunto de rotas, originando e terminando no depósito, que atenda todos os clientes com o menor custo possível. O primeiro objetivo é minimizar o número de veículos necessários e, em segundo lugar, busca-se minimizar o tempo total das rotas. A população inicial é gerada a partir da

heurística de inserção I1 de Solomon (1987). O algoritmo foi aplicado aos conjuntos de problemas R1, R2, C1, C2, RC1 e RC2 de Solomon. Segundo os autores, o algoritmo é competitivo com os melhores resultados encontrados na literatura, tanto em relação ao número de rotas quanto ao tempo total de rota.

2.3.3 PROBLEMA DE ROTEIRIZAÇÃO DE VEÍCULOS COM JANELA DE TEMPO FLEXÍVEL (VEHICLE ROUTING PROBLEM WITH SOFT TIME WINDOWS – VRPSTW)

a) MÉTODOS HEURÍSTICOS

Koskosidis, Powell e Solomon (1992) propuseram uma extensão do método *agrupa-roteiriza* do algoritmo de Fisher e Jaikumar (1981) para o problema de roteirização e programação de veículos com janelas de tempo flexíveis e frota homogênea. Os autores utilizam a mesma formulação do VRP de Fisher e Jaikumar, acrescentando as restrições de janela de tempo, que podem ser violadas mediante o pagamento de penalidades. O objetivo do modelo é minimizar a distância total percorrida e as penalidades pagas com a violação das janelas de tempo. O problema original é dividido em dois subproblemas. O primeiro é o problema de alocação e agrupamento, e o segundo corresponde ao problema do caixeiro viajante com restrição de janela de tempo.

A restrição de janela de tempo é então relaxada, podendo ser quebrada mediante o pagamento de penalidades. O problema passa a ser formulado como um problema não-linear generalizado de alocação (NGAP). Como na heurística de Fisher e Jaikumar, aproxima-se a função custo, resultando na formulação do problema generalizado de alocação (GAP). A solução é obtida resolvendo o GAP, que irá gerar k agrupamentos. Para cada agrupamento, resolve-se o problema do caixeiro viajante com janela de tempo.

Koskosidis, Powell e Solomon aplicaram o algoritmo ao conjunto de problemas R1, C1 e RC1 de Solomon (1987). Os resultados foram, em média, 6% menores, porém, no trabalho de Solomon não se permitiu quebra na janela de tempo. Considerando somente os problemas com 100% de clientes com janela de tempo, observou-se uma redução de apenas 0,81% em relação aos resultados de Solomon (1987), sendo que algumas janelas de tempo ainda foram quebradas.

Balakrishman (1993) apresenta três heurísticas para o problema de roteirização e programação de veículos com janela de tempo flexível (VRPSTW). O problema tem como característica frota de veículos homogênea e há restrições de capacidade dos veículos, janelas de tempo flexíveis e duração máxima da jornada de trabalho. O número de veículos é ilimitado e, portanto, automaticamente determinado pela heurística. O modelo minimiza o tempo total das rotas e as penalidades de adiantamento e atraso pela violação da janela de tempo. A primeira heurística é baseada no vizinho mais próximo, a segunda, na heurística das economias e a terceira, na heurística de tempo-espacó encontrada no trabalho de Sexton e Choi (1986). O autor mostra que, permitindo-se a violação das janelas de tempo, pode-se diminuir significativamente o número de veículos necessários e/ou a distância total e a duração total das rotas. Balakrishnan aplica os três algoritmos no conjunto de problemas proposto por Solomon (1987). Os melhores resultados foram obtidos para o conjunto de problemas R1, a partir da heurística do vizinho mais próximo. Para problemas onde as janelas de tempo são rígidas, os procedimentos não parecem adequados (BREJON, 1998).

b) BUSCA TABU

Taillard et al. (1997) implementaram a metaheurística busca tabu em um problema de roteirização e programação de veículos com restrição de janela de tempo flexível (VRPSTW) e frota de veículos homogênea. O objetivo é minimizar a distância total percorrida, além da soma dos atrasos. A solução inicial foi gerada através de uma adaptação da heurística de inserção I1 de Solomon (1987). Um conjunto de soluções foi gerado introduzindo um componente estocástico na heurística I1. Para reduzir o tempo computacional e intensificar a busca, cada solução é decomposta em subproblemas (TAILLARD, 1993). O método foi aplicado ao conjunto de problemas proposto por Solomon (1987) e comparado com os melhores resultados encontrados na literatura. Segundo os autores, a metaheurística apresentada superou as melhores soluções em diferentes classes de problemas.

Lau, Sim e Teo (2003) implementaram busca tabu para um problema de roteirização com janelas de tempo (VRPTW) e número de veículos limitado. Em função da limitação do número de veículos, a solução factível é obtida considerando que alguns clientes não são atendidos e/ou que as restrições de janelas de tempo poderão ser relaxadas mediante o pagamento de penalidades (VRPSTW). O primeiro objetivo é maximizar o número de clientes

atendidos e, em segundo lugar, minimiza-se a distância total percorrida. O algoritmo foi aplicado ao conjunto de problemas proposto por Solomon (1987). Segundo os autores, o algoritmo gerou soluções próximas às melhores soluções encontradas na literatura, com um bom desempenho computacional.

2.3.4 PROBLEMA DE ROTEIRIZAÇÃO DE VEÍCULOS COM ENTREGAS FRACIONADAS (VEHICLE ROUTING PROBLEM WITH SPLIT DELIVERIES – VRPSD)

a) MÉTODOS EXATOS

Dror, Laporte e Trudeau (1994) apresentaram a formulação de um modelo de programação linear inteira para o VRPSD com frota homogênea e descreveram novas classes de desigualdades válidas para o problema. Os autores também propuseram uma heurística como limitante superior e aplicaram um algoritmo *branch and bound* para o problema.

Belenguer, Martinez e Mota (2000), através de uma nova classe de desigualdades válidas, propuseram um algoritmo de plano de corte e um algoritmo *branch and bound* para um problema de roteirização de veículos com entrega fracionada (VRPSD). O algoritmo de plano de corte foi aplicado para resolução de problemas de pequenas instâncias, e o algoritmo *branch and bound* para problemas maiores. O problema tem como característica frota de veículos homogênea, a demanda de cada cliente é menor que a capacidade dos veículos e a quantidade entregue a cada cliente é inteira. O objetivo é determinar um conjunto de rotas que minimize a distância total percorrida, de forma que a demanda de todos os clientes seja satisfeita, respeitando as restrições de capacidade do veículo.

a) MÉTODOS HEURÍSTICOS

O problema de roteirização de veículos com entregas fracionadas foi introduzido na literatura por Dror e Trudeau (1989, 1990) que apresentaram a formulação matemática do problema e analisaram as economias que podem ser geradas, quando se permite que um cliente possa ser atendido por mais de um veículo, tanto em relação ao número de veículos quanto à distância total percorrida. Estas economias foram significativas para problemas com demanda acima de 10% da capacidade do veículo. O problema tem como característica frota

de veículos homogênea e demanda de cada cliente menor que a capacidade dos veículos. O objetivo é minimizar a distância total percorrida, de forma que a restrição de capacidade dos veículos seja respeitada. O algoritmo implementado é baseado em busca local e é dividido em duas fases. A fase 1 consta das seguintes etapas:

1. Construção de uma solução viável para um problema de roteirização de veículos (VRP), com apenas um veículo por cliente. O algoritmo é uma variação da heurística de Clarke e Wright (1964).
2. Troca de nós: baseada na troca de um e dois nós entre rotas.
3. Heurística de melhoria: troca de arcos 2-*opt* entre rotas.

Na fase 2 temos:

4. Trocas *k-split*: Considera-se $d_{p(k+1)}$ a demanda de um cliente p que está inserido na rota $k+1$ e $s_v, v = 1, \dots, NV$ a capacidade ociosa do veículo v . Se $s_1 + s_2 + \dots + s_k \geq d_{p(k+1)}$ e houver economia, então a demanda do cliente p pode ser eliminada da rota $k+1$ e adicionada às k rotas e o custo total é reduzido.
5. Adição de rotas: Considera-se um cliente p que está inserido em duas rotas ($k = 1$ e $k = 2$). Verifica-se se, com a adição de uma nova rota e eliminação do cliente p das rotas $k = 1$ e $k = 2$, o custo total é reduzido. Se isto ocorrer, o movimento é efetuado.
6. Troca de nós e heurística de melhoria (semelhante à fase 1)

Segundo Dror e Trudeau, o VRPSD é uma relaxação do problema clássico de roteirização de veículos, mas permanece *NP*-completo. Os autores demonstraram que, quando as distâncias satisfazem a desigualdade do triângulo, existe uma solução ótima para VRPSD, de modo que nenhum par de rotas tenha dois ou mais vértices em comum.

Frizzell e Giffin (1992) desenvolveram três heurísticas para um problema de roteirização de veículos com entregas fracionadas (VRPSD), baseadas no vizinho mais próximo, em agrupamentos de clientes e introduzindo custos no caso de entregas fracionadas. As heurísticas foram aplicadas em um conjunto com 1050 problemas. O problema tem como característica frota homogênea e demanda dos clientes menor que a capacidade dos veículos. O objetivo do modelo é minimizar a distância total percorrida, respeitando a restrição de

capacidade dos veículos. Os autores concluem que problemas com um grande número de clientes e uma alta taxa de demanda são viáveis para entregas fracionadas.

Archetti, Savelsbergh e Speranza (2003a) analisaram o comportamento do pior-caso (*worst-case analysis*) para o problema de roteirização de veículos com entregas fracionadas (VRPSD). Os autores demonstraram que a opção de entregas fracionadas pode gerar economias de no máximo 50%, no caso em que a demanda dos clientes é menor que a capacidade dos veículos. Estudaram também, o caso em que a demanda de cada cliente pode ser maior que a capacidade do veículo, onde cada cliente deve ser visitado um número mínimo de vezes. Neste caso, também foram geradas economias de no máximo 50%. Os autores também analisaram o desempenho de simples heurísticas, no caso em que a demanda pode ser maior que a capacidade dos veículos, enviando veículos com carga completa até que a demanda seja menor ou igual à capacidade do caminhão. Finalmente, foram analisados casos com demanda discreta e capacidade dos veículos pequena.

b) BUSCA TABU

Archetti, Hertz e Speranza (2003b) descreveram um algoritmo tabu para um problema de roteirização de veículos com entrega fracionada (VRPSD), considerando uma frota de veículos homogênea. A demanda de cada cliente pode ser maior do que a capacidade do veículo. O objetivo é determinar um conjunto de rotas que atenda a demanda dos clientes e respeite a restrição de capacidade dos veículos, de forma a minimizar a distância total percorrida. Os autores apresentam a formulação matemática do problema. O algoritmo proposto foi dividido em três fases. Na fase 1 é construída uma solução inicial. A fase 2 consiste na implementação do algoritmo busca tabu. Já na fase 3 aplica-se um método de melhoria na solução final encontrada na fase 2.

Para a fase 1, o algoritmo cria inicialmente o número máximo de rotas diretas, enquanto a demanda de cada cliente for maior que a capacidade do veículo. O próximo passo da fase 1 é a construção de múltiplos caixeiros viajantes, através do algoritmo GENIUS de Gendreau, Hertz e Laporte (1992). O algoritmo GENIUS é composto de duas etapas: a primeira (GENI) é um método de inserção e a segunda (US) é um procedimento de melhoria. A fase 2 consiste na implementação do algoritmo busca tabu. Segundo os autores, o algoritmo é muito simples, fácil de implementar e há apenas dois parâmetros a serem ajustados: a

duração da lista tabu e o número máximo de iterações sem que haja melhoria da função objetivo. O método de melhoria implementado na fase 3 consiste em duas etapas.

Conforme descrito em Dror e Trudeau (1989, 1990), se as distâncias satisfazem a desigualdade do triângulo, existe uma solução ótima para VRPSD, onde cada par de rotas tem no máximo um vértice t em comum ($t \leq 1$). Assim, na primeira etapa da fase 3, para um determinado par de rotas, se as distâncias satisfazem a desigualdade do triângulo e há mais de um vértice em comum ($t > 1$), eliminam-se $t - 1$ vértices em comum de uma das rotas. Suponhamos duas rotas r_w e r_{w+1} que possuem dois clientes i_1 e i_2 em comum. A demanda do cliente i_1 é eliminada da rota r_w e transferida para a rota r_{w+1} . A segunda etapa da melhoria é aplicada para cada uma das rotas obtidas (trocas na mesma rota), incorporando o método US.

2.3.5 PROBLEMA DE ROTEIRIZAÇÃO DE VEÍCULOS COM JANELAS DE TEMPO E ENTREGAS FRACIONADAS (VEHICLE ROUTING PROBLEM WITH TIME WINDOWS AND SPLIT DELIVERIES – VRPTWSD)

a) MÉTODO HEURÍSTICO

Frizzell e Giffin (1995) apresentaram a formulação matemática para um problema de roteirização e programação de veículos com entregas fracionadas e restrição de janela de tempo (VRPTWSD). O problema tem como característica frota homogênea, demanda menor que a capacidade dos veículos e restrições de capacidade dos veículos e janelas de tempo. Os autores desenvolveram uma heurística construtiva baseada na urgência dos clientes e duas heurísticas de melhoria; uma delas permite troca de clientes da mesma rota e a outra, troca de clientes entre rotas. As heurísticas foram aplicadas em diversos problemas encontrados na literatura.

b) BUSCA TABU

Ho e Haugland (2004) implementaram a metaheurística busca tabu para o VRPTWSD. A demanda dos clientes pode exceder a capacidade do veículo. A frota de veículos é homogênea e os roteiros iniciam e retornam ao depósito no horário máximo estabelecido. O

objetivo do modelo é minimizar o número de veículos e a distância total percorrida. O modelo foi aplicado em um problema com 100 clientes.

O algoritmo é dividido em três fases. A primeira fase constrói uma solução inicial para o problema, através de um algoritmo de inserção seqüencial, que minimiza a soma dos tempos de viagem e espera dos clientes. Caso a demanda de um determinado cliente a ser inserido exceda a capacidade do veículo, a demanda é adicionada enquanto houver capacidade e a demanda remanescente é adicionada a uma nova rota. A segunda fase corresponde à implementação da metaheurística busca tabu. A vizinhança é baseada em movimentos de trocas entre operadores, como:

1. Inserção: inserir um cliente i de uma rota R_k para uma rota R_l (ver figura 2.5).
2. Realocação de clientes com demandas fracionadas: o cliente i , que era atendido pelas rotas R_k e R_l , é removido da rota R_k , e passa a ser atendido apenas pela rota R_l . Já o cliente j , que era atendido apenas pela rota R_l , passa a ser atendido também pela rota R_k (ver figura 2.6).
3. Troca de clientes entre rotas (ver figura 2.7)
4. 2-opt: Troca de 2 arcos entre rotas (ver figura 2.8)
5. Economia: eliminar rotas com poucos clientes e inserir em outras rotas, através do passo 1.
6. Realocação de clientes na mesma rota, através do algoritmo US de Gendreau et al. (1992).

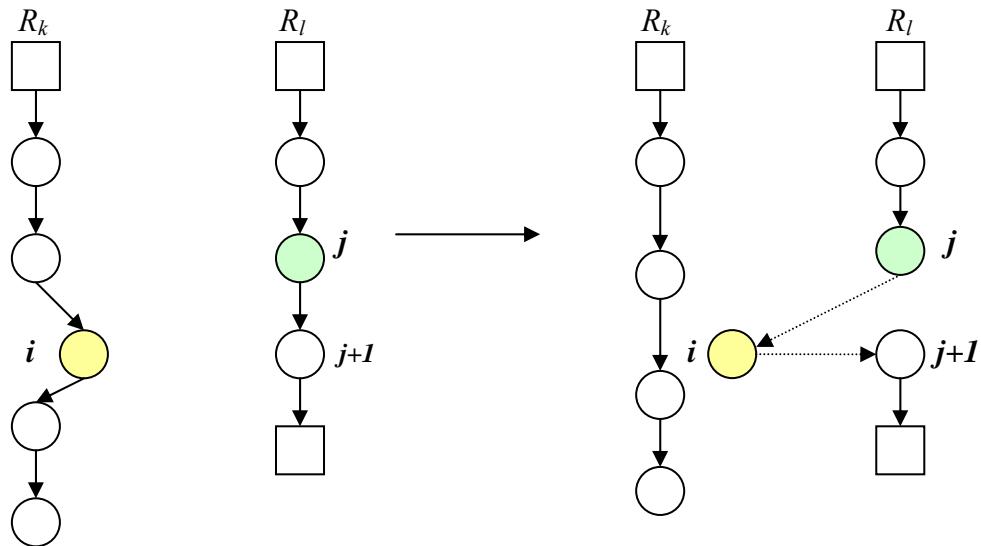
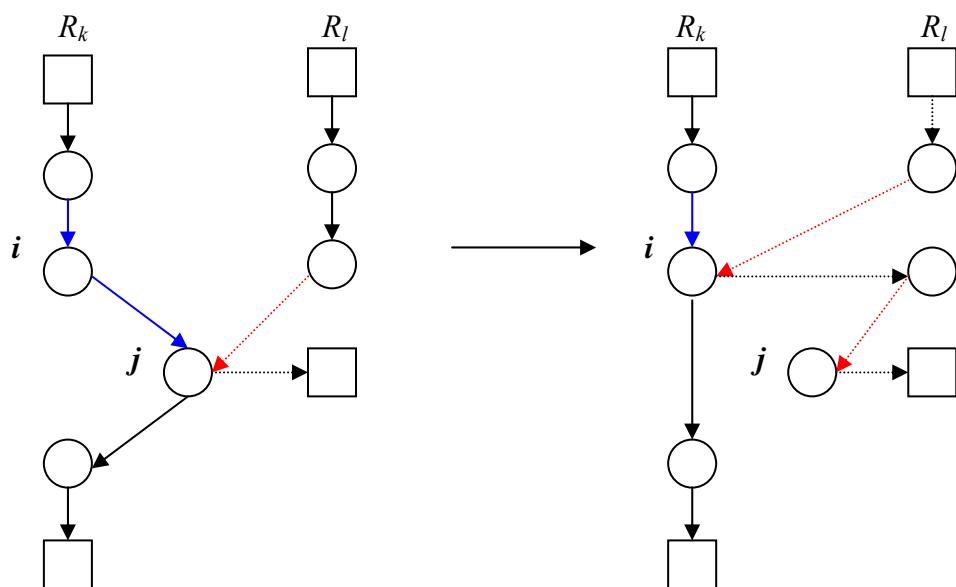
Figura 2.5 - Inserção**Figura 2.6 - Troca de clientes fracionados**

Figura 2.7 - Troca de clientes entre rotas

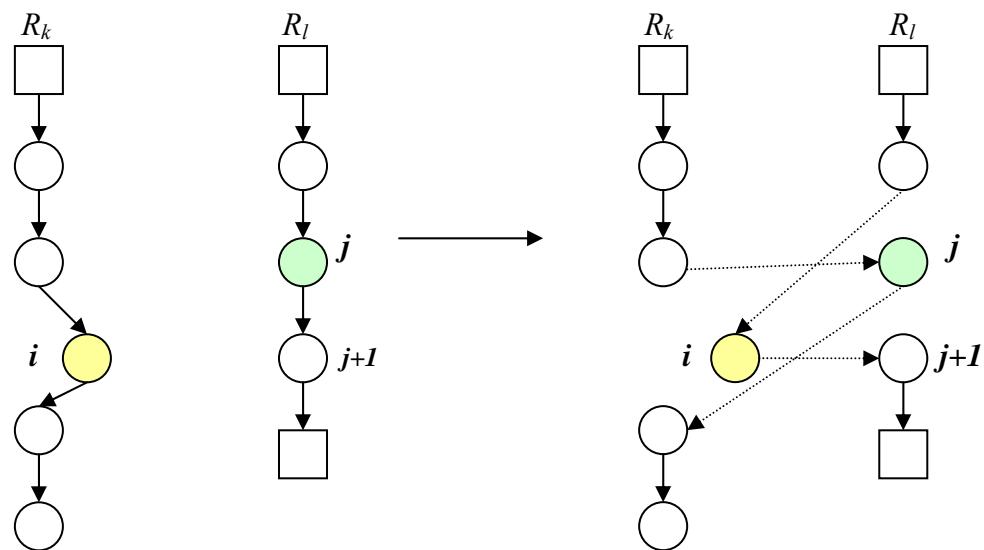
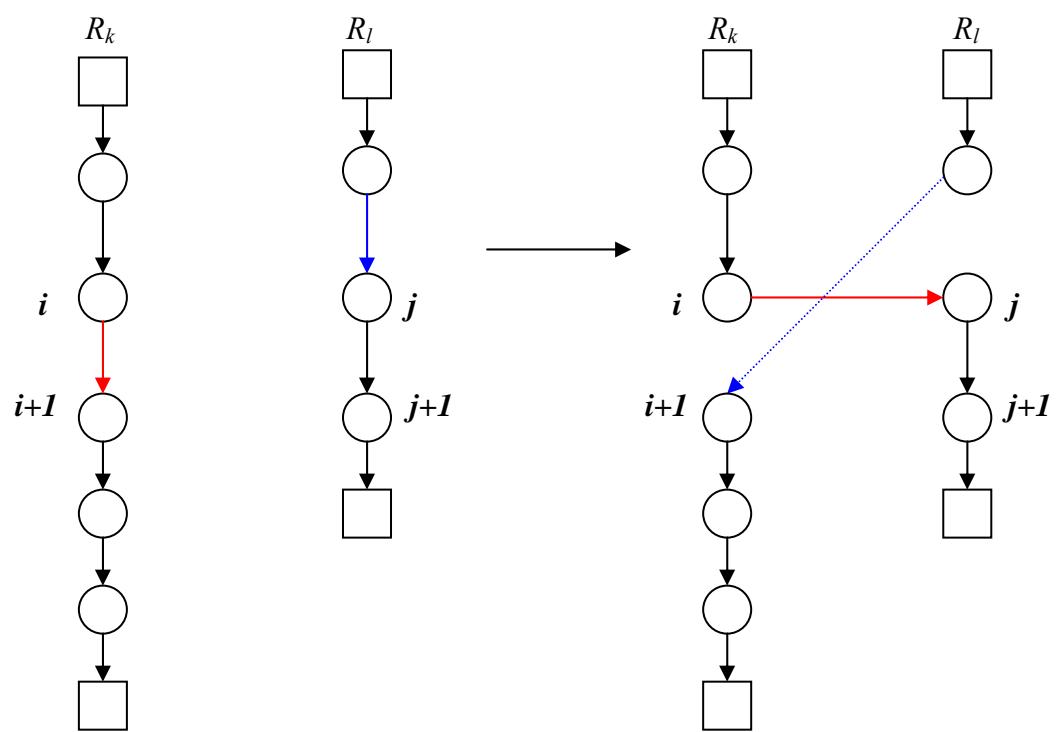


Figura 2.8 - Troca de arcos



Finalmente, a terceira fase consiste em aplicar melhoria nas melhores soluções encontradas, utilizando o algoritmo US e o critério de Realocação.

Os autores demonstraram que a propriedade de desigualdade do triângulo de Dror e Trudeau (1989, 1990) também vale para VRPTWSD:

Preposição. Se VRPTWSD tem uma solução factível, e $\{t_{ij}\}_{i,j \in N}$ e $\{c_{ij}\}_{i,j \in N}$ satisfazem a desigualdade do triângulo, então o problema tem uma solução ótima onde nenhum par de rotas tem mais de um cliente em comum.

Com base no conjunto de problemas original proposto por Solomon (1987), os autores modificaram as demandas de modo que elas possam ser maiores que a capacidade dos veículos, permitindo entregas fracionadas. Neste caso, o algoritmo foi aplicado tanto para o VRPTWSD como o VRPTW, com o objetivo de comparar os resultados de problemas de roteirização com janelas de tempo com e sem entregas fracionadas. Considerando as 56 classes de problemas de Solomon, os custos do VRPTWSD foram maiores que do VRPTW apenas para duas classes de problemas. Por outro lado, o tempo computacional foi bastante superior para problemas com rotas fracionadas, principalmente no caso de demandas elevadas.

As heurísticas também foram aplicadas ao conjunto de problemas original de Solomon (VRPTW). Os resultados do algoritmo foram comparados com os melhores resultados obtidos na literatura, para cada uma das 56 classes e, segundo os autores, superaram ou igualaram as melhores soluções em 13 problemas.

Cabe lembrar que uma das heurísticas construtivas implementadas neste trabalho é uma adaptação do algoritmo de inserção seqüencial de Ho e Haugland (2004).

2.3.6 PROBLEMA DE ROTEIRIZAÇÃO DE VEÍCULOS COM DEMANDA EM ARCOS, JANELAS DE TEMPO E ENTREGAS FRACIONADAS (*CAPACITATED ARC ROUTING PROBLEM WITH TIME WINDOWS AND SPLIT DELIVERIES – CARPTWSD*)

a) MÉTODO HEURÍSTICO

Mullaseril, Dror e Leung (1997) implementaram uma adaptação da heurística proposta por Dror e Trudeau (1990) para um problema de roteirização de veículos com demanda em arcos, janelas de tempo e entregas fracionadas (CARPTWSD). A frota de veículos é homogênea e há restrições de janelas de tempo e capacidade dos veículos. Da mesma forma, primeiramente foram geradas soluções factíveis com apenas uma entrega e, a partir daí, aplicaram-se heurísticas para gerar e melhorar (troca de arcos, trocas de clientes com demandas fracionadas, adição de rotas) soluções com entregas fracionadas. O objetivo do modelo é minimizar a distância total percorrida. Os autores mostraram que a opção de mais de uma entrega reduziu a distância total percorrida em 4 dos 5 casos analisados. Em dois casos a melhoria foi acima de 10%.

2.3.7 PROBLEMA DE ROTEIRIZAÇÃO DE VEÍCULOS COM FROTA HETEROGENEIA FIXA (*HETEROGENEOUS FIXED FLEET VEHICLE ROUTING PROBLEM – HFFVRP*)

a) MÉTODOS HEURÍSTICOS

Taillard (1999) propôs uma heurística baseada no método de geração de colunas para resolução do problema de roteirização de veículos com frota heterogênea e limitada. Os custos variáveis são dependentes do tipo de veículo, ou seja, diferem em função do tamanho de veículo. O objetivo é determinar os roteiros de entrega que minimizam a soma dos custos fixos e dos custos de viagem. A geração de colunas é baseada em um procedimento de memória adaptiva de Taillard (1994) para implementação de busca tabu. Segundo o autor, o método também é capaz de resolver o problema de dimensionamento e roteirização de uma frota heterogênea de veículos (FSMVRP). Taillard gerou um conjunto de instâncias para o problema de roteirização de veículos com frota heterogênea fixa (HFFVRP). Segundo Taillard, o método é robusto e eficiente, particularmente para instâncias de problemas de

médio e grande porte.

b) METAHEURÍSTICA BATA

Tarantilis, Kiranoudis e Vassiliadis (2004) implementaram uma nova metaheurística denominada BATA (*Back-Tracking Adaptative Threshold Accepting*) para o problema de roteirização de veículos com frota de veículos heterogênea fixa (HFFVRP). A idéia fundamental desta classe de algoritmos é permitir movimentos que melhorem o valor da função objetivo, de forma a escapar de mínimos locais. O número de veículos de cada tipo é limitado e os custos variáveis diferem em função do tipo de veículo. O algoritmo foi aplicado para resolução de oito instâncias de problemas de Taillard (1999) e, segundo os autores, o método é robusto e eficiente, pois superou os melhores resultados obtidos por Taillard na maioria dos problemas.

2.3.8 PROBLEMA DE ROTEIRIZAÇÃO DE VEÍCULOS COM FROTA HETEROGÊNEA FIXA E RESTRIÇÕES DE JANELAS DE TEMPO (*HETEROGENEOUS FIXED FLEET VEHICLE ROUTING PROBLEM WITH TIME WINDOWS – HFFVRPTW*)

a) MÉTODOS HEURÍSTICOS

Conforme descrito no item 2.3.2, Cunha (1997) implementou, além das heurísticas aplicadas ao VRPTW com frota homogênea, a heurística de agrupamento e alocação seqüencial heurísticas para um problema de roteirização de veículos com frota heterogênea fixa e restrições de janela de tempo, capacidade dos veículos e duração máxima da jornada de trabalho (HFFVRPTW). A estratégia de solução é baseada na relaxação lagrangiana e um algoritmo de etiquetamento permanente. A heurística de agrupamento e alocação seqüencial foi aplicada em um problema real de distribuição com um total de 136 entregas, resultando em uma redução da distância total percorrida, dos custos operacionais e do número de veículos necessário.

b) METAHEURÍSTICA

Rochat e Semet (1994) implementaram a metaheurística busca tabu para um problema

de distribuição de rações de animais com frota heterogênea fixa e restrições de janela de tempo, duração da jornada de trabalho, capacidade dos veículos e restrições quanto ao tipo de veículo para atendimento de alguns clientes. A solução inicial é baseada na heurística de inserção de Solomon (1987). O algoritmo busca tabu utiliza estratégias de relaxação das restrições e intensificação. Através da relaxação das restrições, é possível expandir o espaço de soluções. Segundo os autores, o modelo é capaz de gerar boas soluções em um tempo computacional razoável. As soluções do modelo foram comparadas com a solução atual da empresa e, segundo os autores, verificou-se que a distância total percorrida foi reduzida significativamente com a aplicação do método.

2.3.9 PROBLEMA DE DIMENSIONAMENTO E ROTEIRIZAÇÃO DE UMA FROTA HETEROGÊNEA DE VEÍCULOS (*FLEET SIZE AND MIX VEHICLE ROUTING PROBLEM – FSMVRP*)

a) MÉTODOS HEURÍSTICOS

Golden et al. (1984) implementaram heurísticas baseadas no método das economias de Clarke e Wright (1964) e no roteiro gigante (roteiriza depois agrupa) para o problema de dimensionamento e roteirização de uma frota heterogênea de veículos (*Fleet Size and Mix Vehicle Routing Problem – FSMVRP*). O objetivo é minimizar a soma dos custos fixos e variáveis dos veículos. Como os custos de viagem independem do tipo de veículo utilizado, os custos variáveis de roteirização são proporcionais à distância total percorrida. Foram implementadas também heurísticas de melhorias baseadas na troca de arcos do tipo 2-opt e 3-opt. As heurísticas de Golden et al. buscam superar a deficiência da heurística de economias para problemas com frota heterogênea, substituindo as distâncias por custos variáveis unitários multiplicados pela distância e adicionando o custo fixo do menor veículo capaz de atender a demanda solicitada. Os autores geraram um conjunto de instâncias de problemas para o FSMVRP.

Desrochers e Verhoog (1991) estudaram uma extensão da heurística de economias proposta por Golden et al. (1984) para o problema de dimensionamento e roteirização de uma frota heterogênea. O objetivo é determinar a composição da frota e um conjunto de rotas que minimizem os custos fixos dos veículos (custo de aquisição e manutenção da frota) e os custos de roteirização que são proporcionais à distância total percorrida, respeitando as

restrições de capacidade dos veículos e duração máxima da jornada de trabalho. As economias eram baseadas em um problema de emparelhamento ponderado (*weighted matching problem*), ou seja, na fusão de sucessivas rotas. A cada iteração, a melhor fusão é selecionada através do problema de emparelhamento ponderado. Ao invés de unir os pontos em ordem decrescente de economias, os autores propuseram resolver um problema de designação em grafo bipartido, considerando-se todas as possibilidades de combinação de pontos e roteiros. A idéia é encontrar a solução do problema de designação que maximize as economias totais, dadas pela soma das economias das rotas unidas (TEIXEIRA; CUNHA, 2002). Segundo os autores, o método apresenta bons resultados, é de fácil implementação e é menos míope do que outras heurísticas de economia.

Gouvêa (1992) propôs uma extensão do método de partição múltipla do roteiro gigante proposto por Golden et al. (1984) para o problema de dimensionamento e roteirização de uma frota heterogênea de veículos, com restrições de capacidade e duração máxima da jornada de trabalho. Gouvêa propôs também um procedimento de alocação de frotas baseado em inserção seqüencial e de melhoria das soluções obtidas através de trocas de arcos do tipo *2-opt*. Os custos variáveis diferem em função do tipo de veículo.

Salhi e Rand (1993) desenvolveram uma heurística de sete fases para o FSMVRP. O objetivo é minimizar a soma dos custos fixos dos veículos e dos custos de roteirização que são proporcionais à distância total percorrida. O algoritmo é baseado nos seguintes movimentos: 1) combinação de rotas; 2) eliminação de rotas onde os clientes eliminados são inseridos em outras rotas existentes; 3) inserção de clientes de uma determinada rota em outra; 4) fracionamento de grandes rotas em rotas menores; 5) troca de clientes entre rotas. O procedimento foi aplicado no conjunto de problemas proposto por Golden et al. (1984). Segundo os autores, o algoritmo superou os melhores resultados obtidos na literatura em 7 casos.

Teixeira e Cunha (2002) implementaram três heurísticas com base no método de economias de Clarke e Wright (1964), nos trabalhos de Golden et al. (1984) e Desrochers e Verhoog (1991), para o problema de dimensionamento e roteirização de uma frota heterogênea (FSMVRP) com restrição de capacidade dos veículos.

A primeira heurística, chamada heurística básica, é baseada no trabalho de Desrochers e Verhoog (1991) e é uma generalização do método de economias de Clarke e Wright (1964). Porém, neste caso, o problema de designação em grafo bipartido foi modelado como um Problema de Circulação com Custo Mínimo e resolvido através do algoritmo *Out-of-Kilter*. Os resultados mostraram que o tempo de processamento desta heurística foi elevado.

Com o objetivo de melhorar o desempenho computacional da heurística básica e da heurística de Desrochers e Verhoog (1991), Teixeira e Cunha propuseram uma heurística modificada denominada heurística híbrida. A cada solução do problema de designação unem-se todas as rotas que fazem parte da solução, ao invés de unir apenas o par de maior economia, como proposto pelos autores. A terceira heurística, chamada heurística de seleção direta, é uma adaptação da heurística de economias proposta por Golden et al. (1984).

Segundo os autores, a heurística híbrida gerou soluções de melhor qualidade em relação à heurística de Desrochers e Verhoog (1991) e à heurística básica. Além disso, todas apresentaram resultados melhores que a heurística de economias generalizada proposta por Golden et al. (1984), em que os pontos são unidos seqüencialmente, em ordem decrescente de economias. A heurística de seleção direta apresentou o pior resultado. Apesar de gerar soluções de alta qualidade, o tempo de processamento da heurística híbrida ainda é muito elevado, o que dificulta a aplicação destas heurísticas a problemas reais.

b) BUSCA TABU

Gendreau et al. (1999) implementaram a metaheurística busca tabu para o problema de dimensionamento e roteirização de uma frota heterogênea de veículos. A frota de veículos é ilimitada e os custos variáveis são dependentes do tipo de veículo. A solução inicial é obtida através do algoritmo GENI e, no final desta fase, cada rota é reotimizada através da fase US. A busca tabu é baseada em um procedimento de memória adaptiva (*adaptive memory procedure – AMP*) desenvolvida por Rochat e Taillard (1995). O algoritmo foi implementado em 12 instâncias de Golden et al. (1984) e em 8 instâncias de Taillard (1999). Segundo os autores, o algoritmo gerou soluções de alta qualidade, superando, em vários casos, as melhores soluções obtidas na literatura.

Wassan e Osman (2002) implementaram a metaheurística busca tabu para o FSMVRP.

O objetivo é determinar a composição ótima da frota e os roteiros de entrega de forma a minimizar a soma dos custos fixos e variáveis dos veículos. A frota de veículos é ilimitada e os custos variáveis são dependentes do tipo de veículo. Foram implementados quatro mecanismos de vizinhança, baseados na troca entre operadores. Os autores mostraram que o desempenho do algoritmo não depende do tamanho da vizinhança e sim, da estratégia utilizada na vizinhança e do procedimento de busca. O algoritmo foi aplicado no conjunto de 20 instâncias de Golden et al. (1984) e em 8 instâncias de Taillard (1999). Segundo os autores, o algoritmo se mostrou eficiente tanto em termos computacionais como de qualidade da solução, superando os melhores resultados obtidos para algumas instâncias de problemas.

2.3.10 PROBLEMA DE DIMENSIONAMENTO E ROTEIRIZAÇÃO DE UMA FROTA HETEROGRÉNEA DE VEÍCULOS COM RESTRIÇÕES DE JANELAS DE TEMPO (*FLEET SIZE AND MIX VEHICLE ROUTING PROBLEM WITH TIME WINDOWS – FSMVRPTW*)

a) MÉTODOS HEURÍSTICOS

Liu e Shen (1999) implementaram uma heurística de inserção seqüencial baseada no algoritmo de economias para um problema de dimensionamento e roteirização de uma frota heterogênea de veículos com janelas de tempo (FSMVRPTW). O objetivo do FSMVRPTW é minimizar a soma dos custos fixos e dos custos de roteirização dos veículos que são proporcionais à distância total percorrida. O algoritmo foi implementado em 168 problemas que são adaptações do conjunto de instâncias de Solomon (1987) para frota heterogênea. A heurística também foi implementada no conjunto de problemas proposto por Golden et al. (1984) para FSMVRP. Segundo os autores, a heurística apresentou bons resultados para ambos os casos.

Dullaert et al. (2002) implementaram três heurísticas para o problema de dimensionamento e roteirização de uma frota heterogênea de veículos com restrições de janelas de tempo (FSMVRPTW). Os algoritmos são extensões da heurística de inserção seqüencial I1 de Solomon e, além do acréscimo de distância e tempo, considera-se um custo baseado nas economias de Golden et al. (1984) para inserção de um cliente. O objetivo é minimizar a soma dos custos fixos e dos custos de viagem que são proporcionais à distância total percorrida. As heurísticas são implementadas no conjunto de problemas de Liu e Shen

(1999) que são adaptações das instâncias de Solomon (1987). A análise dos resultados é baseada no tempo total das rotas, excluindo o tempo de serviço, e, segundo os autores, a melhoria foi superior a 50% em muitos problemas.

Cabe lembrar que uma das heurísticas construtivas implementadas no presente trabalho é uma adaptação dos algoritmos de Dullaert et al. (2002).

Outras referências sobre problemas de roteirização com frota heterogênea podem ser encontradas em Gheysens et al. (1984), Gheysens et al. (1986), Ferland e Michelon (1988), Salhi et al. (1992), Taillard et al. (1996), Brandão e Mercer (1997).

O quadro 2.2 apresenta as características específicas de cada um dos trabalhos apresentados neste capítulo, como tipo de problema, restrições, método de solução e função objetivo.

Quadro 2.2 - Sumário dos principais métodos de solução para VRP, VRPTW, VRPSD, HFVRP e suas extensões.

Referência Bibliográfica	Tipo de Problema	Restrições	Método de Solução	Função objetivo
Christofides et al. (1981)	VRP	capacidade dos veículos duração máxima jornada	método exato relaxação lagrangiana program. dinâmica relaxada	min distância total
Achuthan et al. (2003)	VRP	capacidade dos veículos	método exato branch and cut	min distância total
Clarke e Wright (1964)	VRP	capacidade dos veículos	heurística construtiva economias	min distância total
Gillet e Miller (1974)	VRP	capacidade dos veículos duração máxima jornada	heurística construtiva agrupa-roteiriza	min distância total
Fisher e Jaikumar (1981)	VRP	capacidade dos veículos	heurística construtiva agrupa-roteiriza	min distância total
Desrochers e Verhoog (1989)	VRP	capacidade dos veículos duração máxima jornada	heurística construtiva economias	min distância total
Taillard (1993)	VRP	capacidade dos veículos duração máxima jornada	metaheurística busca tabu	min distância total
Osman (1993)	VRP	capacidade dos veículos	metaheurística híbrida busca tabu e simulated annealing	min distância total
Gendreau et al. (1994)	VRP	capacidade dos veículos duração máxima jornada	metaheurística busca tabu	min distância total
Barbarosoglu e Ozgur (1999)	VRP	capacidade dos veículos duração máxima jornada	metaheurística busca tabu	min distância total
Kelly e Xu (1999)	VRP	capacidade dos veículos	metaheurística busca tabu	min distância total
Baker e Ayeche (2003)	VRP	capacidade dos veículos duração máxima jornada	metaheurística algoritmos genéticos	min distância total

Continuação do Quadro 2.2 - Sumário dos principais métodos de solução para VRP, VRPTW, VRPSD, HFVRP e suas extensões.

Referência Bibliográfica	Tipo de Problema	Restrições	Método de Solução	Função objetivo
Baker (1982)	VRPTW	capacidade dos veículos janelas de tempo	método exato branch and bound com algoritmo de etiquetamento	min tempo total rotas
Kolen et al. (1987)	VRPTW	capacidade dos veículos janelas de tempo	método exato branch and bound	min distância total
Desrochers et al. (1992)	VRPTW	capacidade dos veículos janelas de tempo	método exato geração de colunas	min distância total
Bard et al. (2002)	VRPTW	capacidade dos veículos janelas de tempo	método exato branch and cut	min número de veículos + distância total
Solomon (1986)	VRPTW	capacidade dos veículos janelas de tempo	heurísticas construtivas economias vizinho mais próximo inserção agrupa-roteiriza roteiro gigante heurística de melhoria troca de arcos	min número veículos + distância total + tempo total rotas
Solomon (1987)	VRPTW	capacidade dos veículos janelas de tempo	heurísticas construtivas economias economia c/ limite de espera vizinho mais próximo c/ orientação temporal inserção I1 inserção I2 inserção I3 varredura c/ orient. temporal	min distância total + tempo total rotas
Solomon et al. (1988)	VRPTW	capacidade dos veículos janelas de tempo	heurísticas de melhoria troca de arcos	min distância + tempo total rotas

Continuação do Quadro 2.2 - Sumário dos principais métodos de solução para VRP, VRPTW, VRPSD, HFVRP e suas extensões.

Referência Bibliográfica	Tipo de Problema	Restrições	Método de Solução	Função objetivo
Thompson e Psaraftis (1993)	VRPTW	capacidade dos veículos janelas de tempo	heurística de melhoria transferência cílica	min nº de veículos + tempo total rotas + distância total
	VRP	capacidade dos veículos		min distância total
Potvin e Rousseau (1993)	VRPTW	capacidade dos veículos janelas de tempo	heurísticas construtivas inserção paralela	min distância total + tempo total de espera
Russell (1995)	VRPTW	capacidade dos veículos janelas de tempo duração máxima jornada	heurística construtiva inserção paralela heurística de melhoria troca de nós entre rotas	min número de veículos + tempo total rotas + distância total
Potvin e Rousseau (1995)	VRPTW	capacidade dos veículos janelas de tempo	heurísticas de melhoria 2-opt e troca de arcos	min nº de veículos + tempo total rotas
Cunha (1997)	VRPTW	capacidade dos veículos janelas de tempo duração máxima jornada	heurística relaxação lagrangiana	min distância total + tempo total rotas
Garcia et al. (1994)	VRPTW	capacidade dos veículos janelas de tempo	metaheurística busca tabu	min distância total + tempo total de espera
Potvin et al. (1996)	VRPTW	capacidade dos veículos janelas de tempo	metaheurística busca tabu	min nº de veículos + distância total + tempo total de espera
Badeau et al. (1997)	VRPTW	capacidade dos veículos janelas de tempo	metaheurística busca tabu	min distância total
	VRPSTW	capacidade dos veículos janelas de tempo flexíveis		min distância total + atraso

Continuação do Quadro 2.2 - Sumário dos principais métodos de solução para VRP, VRPTW, VRPSD, HFVRP e suas extensões.

Referência Bibliográfica	Tipo de Problema	Restrições	Método de Solução	Função objetivo
Chiang e Russell (1997)	VRPTW	capacidade dos veículos janelas de tempo duração máxima jornada	metaheurística busca tabu	min nº de veículos + tempo total rotas + distância total
Tan et al. (2001)	VRPTW	capacidade dos veículos janelas de tempo duração máxima jornada	metaheurística busca tabu simulated annealing algoritmos genéticos	min distância
Kontoravdis e Bard (1995)	VRPTW	capacidade dos veículos janelas de tempo	metaheurística GRASP	min número de veículos + distância total
Potvin e Bengio (1996)	VRPTW	capacidade dos veículos janelas de tempo	metaheurística algoritmos genéticos	min nº de veículos + tempo total das rotas
Koskosidis et al. (1992)	VRPSTW	capacidade do veículo janelas de tempo flexíveis	heurística construtiva agrupa-roteiriza	min distância + penalidades
Balakrishnan (1993)	VRPSTW	capacidade do veículo janelas de tempo flexíveis duração máxima jornada	heurísticas construtivas vizinho mais próximo economias tempo-espaco	min tempo total rotas + penalidades
Taillard et al. (1997)	VRPSTW	capacidade dos veículos janelas de tempo flexíveis	metaheurística busca tabu	min distância + penalidades de atraso
Lau et al. (2003)	VRPSTW	capacidade dos veículos janelas de tempo flexíveis	metaheurística busca tabu	max nº clientes atendidos min distância total
Dror et al. (1994)	VRPSD d<Cv	capacidade dos veículos	método exato branch and bound	min distância total
Belenguer et al. (2000)	VRPSD d<Cv	capacidade dos veículos	método exato plano de corte branch and bound	min distância total

Continuação do Quadro 2.2 - Sumário dos principais métodos de solução para VRP, VRPTW, VRPSD, HFVRP e suas extensões.

Referência Bibliográfica	Tipo de Problema	Restrições	Método de Solução	Função objetivo
Dror e Trudeau (1989, 1990)	VRPSD $d < Cv$	capacidade dos veículos	heurística construtiva economias heurísticas de melhoria troca de nós entre rotas troca de arcos trocas k-splits adição de rotas	min distância total
Frizzell e Giffin (1992)	VRPSD $d < Cv$	capacidade dos veículos	heurísticas construtivas vizinho mais próximo agrupamento de clientes	min distância total
Archetti et al. (2003a)	VRPSD $d < Cv$ e $d > Cv$	capacidade dos veículos	heurística	min distância total
Archetti et al. (2003b)	VRPSD $d > Cv$	capacidade dos veículos	metaheurística busca tabu	min distância total
Frizzell e Giffin (1995)	VRPTWSD $d < Cv$	capacidade dos veículos janelas de tempo	heurísticas construtivas look-ahead com base na urgência dos clientes heurísticas de melhoria inserção clientes em outra rota trocas de clientes entre rotas	min distância total
Ho e Haugland (2004)	VRPTWSD $d > Cv$	capacidade dos veículos janelas de tempo horário máximo de retorno	metaheurística busca tabu	min tempo de viagem + espera
Mullaseril et al. (1997)	CARPTWSD $d < Cv$	capacidade dos veículos janelas de tempo	heurística construtiva adaptação de Dror e Trudeau heurísticas de melhoria adaptação de Dror e Trudeau	min distância

Continuação do Quadro 2.2 - Sumário dos principais métodos de solução para VRP, VRPTW, VRPSD, HFVRP e suas extensões.

Referência Bibliográfica	Tipo de Problema	Restrições	Método de Solução	Função objetivo
Taillard (1999)	HFFVRP	capacidade dos veículos	heurística geração de colunas	min custos fixos + custos de viagem
Tarantilis et al. (2004)	HFFVRP	capacidade dos veículos	metaheurística BATA	min custos fixos + custos de viagem
Cunha (1997)	HFFVRPTW	capacidade dos veículos janelas de tempo duração máxima jornada	heurística relaxação lagrangiana	min custos fixos + distância total + tempo total rotas
Rochat e Semet (1994)	HFFVRPTW	capacidade dos veículos janelas de tempo duração máxima jornada tipo de veículo	metaheurística busca tabu	min distância total + tempo total rotas
Golden et al. (1984)	FSMVRP	capacidade dos veículos	heurísticas construtivas economias roteiro gigante heurísticas de melhoria troca de arcos	min custos fixos + distância total
Desrochers e Verhoog (1991)	FSMVRP	capacidade dos veículos duração máxima jornada	heurística construtiva economias	min custos fixos + distância total
Gouvêa (1992)	FSMVRP	capacidade dos veículos duração máxima jornada	heurísticas construtivas economias roteiro gigante heurística de melhoria troca de arcos	min custos fixos + custos de viagem

Continuação do Quadro 2.2 - Sumário dos principais métodos de solução para VRP, VRPTW, VRPSD, HFVRP e suas extensões.

Referência Bibliográfica	Tipo de Problema	Restrições	Método de Solução	Função objetivo
Salhi e Rand (1993)	FSMVRP	capacidade dos veículos duração máxima jornada	heurística de melhoria combinação de rotas eliminação de rotas inserção clientes em outra rota fracionamento de rotas troca de nós entre rotas	min custos fixos + distância
Teixeira e Cunha (2002)	FSMVRP	capacidade dos veículos	heurísticas construtivas economias	min custos fixos + custos de viagem
Gendreau et al. (1999)	FSMVRP	capacidade dos veículos	metaheurística busca tabu	min custos fixos + custos de viagem
Wassan e Osman (2002)	FSMVRP	capacidade dos veículos	metaheurística busca tabu	min custos fixos + custos de viagem
Liu e Shen (1999)	FSMVRPTW	capacidade dos veículos janelas de tempo	heurística construtiva inserção sequencial baseada em economias	min custos fixos + distância total
Dullaert et al. (2002)	FSMVRPTW	capacidade dos veículos janelas de tempo	heurística construtiva inserção sequencial	min custos fixos + distância total

Legenda do tipo de problema

- VRP – Problema clássico de roteirização de veículos
- VRPTW – VRP com janelas de tempo
- VRPSTW – VRP com janelas de tempo flexíveis
- VRPSD – VRP com entregas fracionadas
- VRPTWSD – VRP com janelas de tempo e entregas fracionadas
- CARPTWSD – VRP com demanda arcos, janelas de tempo e entregas fracionadas
- HFFVRP – VRP com frota heterogênea fixa
- HFFVRPTW – VRP com frota heterogênea fixa e janelas de tempo
- FSMVRP – Problema de dimensionamento e roteirização de uma frota heterogênea
- FSMVRPTW – Problema de dimensionamento e roteirização de uma frota heterogênea de veículos com restrições de janelas de tempo

Legenda do método de solução

- Método exato
- Heurística construtiva
- Heurística de melhoria
- Busca tabu
- GRASP
- *Simulated annealing*
- Algoritmos genéticos
- BATA – *Back-Tracking Adaptative Threshold Accepting*

2.4 CONCLUSÃO DA REVISÃO BIBLIOGRÁFICA E CONTRIBUIÇÕES

O problema estudado neste trabalho é uma combinação dos problemas de roteirização com janelas de tempo (VRPTW), problema de roteirização de veículos com entregas fracionadas (VRPSD) e problema de roteirização de veículos com frota heterogênea (HFVRP). Desta forma, o presente capítulo descreveu os principais métodos de solução para estes problemas (e suas extensões), além do problema clássico de roteirização de veículos.

Conforme apresentado no item 2.3, em todos os problemas de roteirização com entregas fracionadas encontrados na literatura (VRPSD, VRPTWSD e CARPTWSD), a frota de veículos é homogênea. Analogamente, em todos os problemas de roteirização de veículos com frota heterogênea estudados, cada cliente pode ser atendido por apenas um veículo, ou seja, não há entregas fracionadas (*split deliveries*). Portanto, o HFVRPTWSD ainda não foi estudado na literatura.

As estratégias de solução propostas consistem na implementação de heurísticas construtivas e da metaheurística *scatter search*. Para a resolução do VRPSD e suas extensões, a única metaheurística encontrada na literatura foi busca tabu. Já para o problema de roteirização com frota heterogênea e suas extensões, foram implementadas apenas as metaheurísticas busca tabu e BATA (*Back-Tracking Adaptative Threshold Accepting*). Portanto, a metaheurística *scatter search* nunca foi implementada para problemas de roteirização de veículos com entregas fracionadas e nem para problemas de roteirização com frota heterogênea.

Podemos concluir, portanto, que o tipo de problema estudado e o método de solução proposto (para problemas de roteirização de veículos com entregas fracionadas e problemas de roteirização com frota heterogênea) são inovadores.

CAPÍTULO 3

SCATTER SEARCH E SUAS APLICAÇÕES EM PROBLEMAS DE ROTEIRIZAÇÃO DE VEÍCULOS

A metaheurística *scatter search* (SS), também conhecida como busca dispersa (BD), será aplicada para resolução do modelo proposto neste trabalho. Este capítulo tem como objetivo descrever os principais conceitos do método e suas aplicações em problemas de roteirização de veículos.

O método é um procedimento metaheurístico para resolver problemas de otimização de natureza combinatória. SS é baseado em formulações e estratégias introduzidas na década de 60, porém os conceitos e princípios fundamentais foram propostos na década de 70 e sua aplicação teve início na década de 90.

Recentes aplicações deste método têm mostrado resultados satisfatórios: roteirização de veículos (REGO; LEÃO, 2005; CORBERÁN et al., 2002; ALEGRE; LAGUNA; PACHECO, 2004), designação quadrática (CUNG et al., 1997), redes neurais (KELLY; RANGASWAMY; XU, 1996), *flow shop scheduling* (JAIN; MEERAN, 2002), ordenação linear (LAGUNA; MARTÍ; CAMPOS, 1999; CAMPOS et al., 2001), otimização não restritiva (FLEURENT et al., 1996), otimização multiobjetivo (LAGUNA; LOURENÇO; MARTÍ, 2000), problema da árvore de cobertura mínima (XU; CHIU; GLOVER, 2000), programação de projetos (YAMASHITA; ARMENTANO; LAGUNA, 2006), etc.

Apresenta similaridades com os algoritmos genéticos e difere no uso de estratégias determinísticas, ao invés de probabilísticas, para atingir diversificação e intensificação, e no tamanho da população. Nos algoritmos genéticos a população é de aproximadamente 100 soluções e a combinação de soluções é feita de forma aleatória. Já na metaheurística SS a população varia em torno de 10 a 20 soluções e novas soluções são criadas com base em um critério sistemático.

SS opera sobre um conjunto de soluções, chamado conjunto de referência, combinando essas soluções para criar novas soluções de modo a melhorar a solução original. É, portanto, um método evolutivo.

A primeira descrição do método foi proposta por Glover em 1977, que apresentou como SS realiza uma exploração sistemática sobre uma série de boas soluções chamadas “conjunto de referência”. Os principais aspectos do trabalho são: o método consiste em combinar duas ou mais soluções do conjunto de referência; atribuem-se pesos durante a combinação de soluções; deve-se realizar combinações convexas e não convexas das soluções; a distribuição dos pontos deve ser dispersa. Glover (1998) apresentou um modelo detalhado do método.

Glover, Laguna e Martí (2000) estudaram novas implementações do método para resolução de problemas de otimização combinatória e descrevem os fundamentos de *scatter search* e *path relinking*. Laguna (2002) implementa a metaheurística SS para resolução de problemas de otimização combinatória e não-linear. O autor ilustra os principais conceitos no contexto do problema clássico de programação inteira. Laguna e Martí (2003) implementam o uso de memória e *path relinking* e descrevem suas aplicações em problemas de otimização combinatória e não-linear. O texto é acompanhado do código-fonte em C para aplicação das técnicas descritas.

Martí, Laguna e Glover (2006) apresentaram os princípios e conceitos fundamentais do método *scatter search*. Os autores descreveram o procedimento básico de SS e, em seguida, estratégias avançadas como atualização dinâmica, reconstrução do conjunto de referência, atualização por qualidade e diversidade, redução do tempo computacional, evitando soluções duplicadas, método de geração de subconjuntos e uso de memória. Os autores também descreveram os princípios de *path relinking*.

Path relinking (PR) foi originalmente proposto como um modelo para integrar estratégias de diversificação e intensificação no contexto da metaheurística busca tabu (GLOVER, 1994; GLOVER; LAGUNA, 1997). Para gerar novas soluções, o modelo inicia por uma das soluções originais (*initiating solution*) e gera um caminho na direção de outras soluções, chamadas soluções guias (*guiding solutions*). *Path relinking* pode ser considerado uma extensão do método de combinação de *scatter search*. Porém, ao invés de gerar novas

soluções diretamente com a combinação de duas ou mais soluções originais, PR gera caminhos entre as soluções selecionadas no espaço de vizinhança analisado (combinações não-lineares). SS e PR diferem, basicamente, na forma como o conjunto de referência é construído, atualizado e melhorado (LAGUNA; MARTÍ, 2003; MARTÍ; LAGUNA; GLOVER, 2006).

O método *scatter search* também pode ser encontrado em Glover (1999), Martí e Laguna (2003), Glover, Laguna e Martí (2003) e Glover, Laguna e Martí (2004). A descrição do método está detalhada a seguir.

3.1 DESCRIÇÃO DO MÉTODO

Scatter search se baseia em combinar as soluções que aparecem no chamado conjunto de referência (*RefSet*). Este conjunto armazena boas soluções que foram encontradas durante o processo de busca. Os principais parâmetros do método *scatter search* estão descritos a seguir:

P = conjunto de soluções geradas pelo método de diversificação

$PSize$ = tamanho da população P

b = tamanho do conjunto de referência (*RefSet*)

b_1 = tamanho do subconjunto das soluções de alta qualidade

b_2 = tamanho do subconjunto de soluções diversas

$MaxIter$ = número máximo de iterações

De acordo com Glover (1998), o método consiste basicamente de cinco elementos descritos a seguir:

1. Método de geração de soluções diversas: o método consiste em gerar um conjunto P com $PSize$ elementos. O objetivo é gerar soluções distantes entre si de forma a garantir a diversidade.

2. Método de melhoria: trata-se de um método de busca local para melhorar as soluções diversas. No caso de soluções infactíveis, a melhoria primeiramente consiste em torná-las factíveis. Caso a solução seja factível, aplica-se a heurística de melhoria.

3. Método de criação e atualização do conjunto de referência: é um método para criar e atualizar o conjunto de referência ($RefSet$). A partir do conjunto de soluções diversas P , extraí-se o conjunto de referência. É dividido em um subconjunto de soluções de alta qualidade ($RefSet_1$) e um subconjunto de soluções diversas ($RefSet_2$). O método SS, portanto, é baseado nas estratégias de intensificação e diversificação.

3.1 Criação do conjunto de referência: o conjunto de referência inicia com as b_1 melhores soluções de P , obtidas após o método de melhoria. As b_2 soluções restantes são extraídas de P com o objetivo de maximizar a distância mínima em relação às soluções incluídas no conjunto de referência. O número de elementos do conjunto de referência (b) varia em torno de 10 a 20 soluções.

3.2 Atualização do conjunto de referência: as soluções resultantes das combinações, no caso de melhoria, substituem soluções do conjunto de referência. Os tipos de atualizações podem ser: a) estática ou dinâmica, b) por qualidade ou diversidade. O tamanho do conjunto de referência (b), das soluções de alta qualidade (b_1) e das soluções diversas (b_2) permanecem sempre constantes, porém, o valor médio das soluções vai melhorando ao longo da busca.

4. Método de geração de subconjuntos: este método especifica a forma pela qual são selecionados os subconjuntos para aplicar o método de combinação.

5. Método de combinação das soluções: busca combinar as soluções do conjunto de referência. Para isso, consideram-se os subconjuntos formados pelo passo 4 e aplica-se o método de combinação. As soluções obtidas desta combinação podem ser imediatamente introduzidas no conjunto de referência (atualização dinâmica) ou armazenadas temporariamente em uma lista até que sejam realizadas todas as combinações e, a partir daí, verificam-se quais soluções entrarão no conjunto (atualização estática). O método de melhoria é aplicado às melhores soluções geradas pelas combinações.

Um algoritmo simplificado do método *scatter search* é apresentado na figura 3.1. O primeiro passo constrói um conjunto inicial de soluções P , com $PSize$ elementos. Para cada uma das soluções diversas geradas aplica-se o método de melhoria (passo 2). O passo 3 consiste na construção do conjunto de referência, selecionando as b_1 melhores soluções de P e as b_2 soluções diversas. No passo 4, as soluções do conjunto de referência são agrupadas em subconjuntos de soluções e as soluções de cada subconjunto são combinadas. Para cada solução combinada aplica-se a heurística de melhoria (passo 5). No caso de atualização dinâmica, as soluções obtidas desta combinação podem ser imediatamente introduzidas no conjunto de referência. No caso de atualização estática, as soluções são armazenadas temporariamente em uma lista até que sejam realizadas todas as combinações e, a partir daí, verificam-se quais soluções entrarão no conjunto. Enquanto houver pelos menos uma nova solução, os passos 4, 5 e 6 são repetidos. No passo 7 reconstrói-se o conjunto de referência ($RefSet$). Isto corresponde a uma iteração do método SS. O algoritmo pára depois de $MaxIter$ iterações (ou qualquer outro critério de parada).

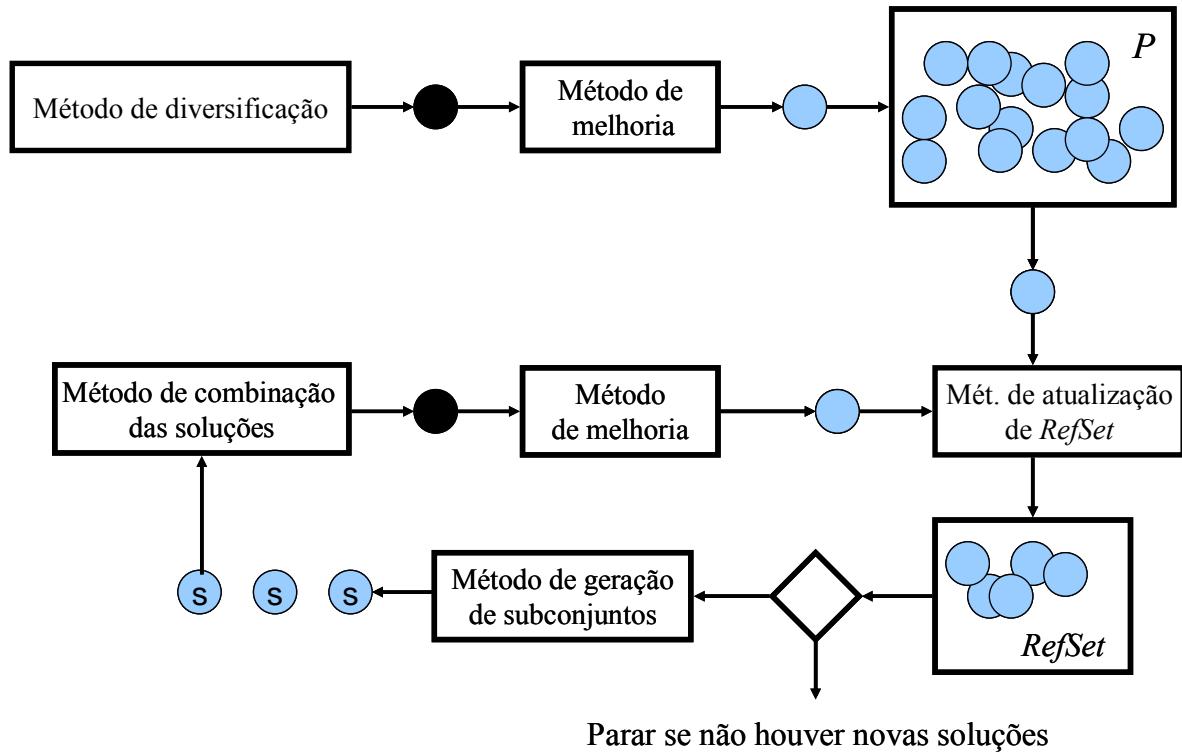
Figura 3.1 - Procedimento *scatter search*

Passo 1: Gerar o conjunto P de $PSize$ soluções diversas (método de diversificação) Passo 2: Aplicar o método de melhoria Passo 3: Construir o conjunto de referência: Selecionar as b_1 melhores soluções e as b_2 soluções mais diversas de P para construir o conjunto de referência ($RefSet$). Número de iterações = 0. $NewSolutions = \text{TRUE}$ Enquanto (número de iterações < $MaxIter$) faça Enquanto houver novas soluções ($NewSolutions$) em $RefSet$ faça Passo 4: Gerar subconjuntos e aplicar o método de combinação Passo 5: Aplicar o método de melhoria para cada solução combinada Passo 6: Atualizar $RefSet$ considerando as novas soluções Fim enquanto Passo 7: Reconstruir $RefSet$: Remover as b_2 piores soluções de $RefSet$. Gerar $PSize$ soluções diversas e aplicar melhoria (passos 1 e 2). Escolher as b_2 soluções mais distantes e adicioná-las ao $RefSet$. Número de iterações = Número de iterações + 1. Fim enquanto

Fonte: Yamashita (2003) e Alegre, Laguna e Pacheco (2004)

A figura 3.2 apresenta o esquema do método *scatter search*.

Figura 3.2 - Esquema do método *scatter search*



Fonte: Martí e Laguna (2003)

A seguir estão detalhados os passos do método *scatter search* com base nos trabalhos de Glover, Laguna e Martí (2004), Yamashita (2003) e Martí, Laguna e Glover (2006).

1. Método de geração de soluções diversas

A primeira fase consiste em gerar soluções diversas. As soluções podem ser geradas através de heurísticas construtivas, podem ser soluções aleatórias ou uma combinação de ambas.

2. Método de melhoria

Para cada uma das soluções diversas geradas na fase 1 aplica-se o método de melhoria. A busca local pode ser baseada em movimentos de troca e inserção. A melhoria também é aplicada após a fase de combinação das soluções.

3. Método de criação e atualização do conjunto de referência

A descrição deste passo foi baseada nos trabalhos de Yamashita (2003) e Yamashita, Armentano e Laguna (2006).

3.1 Criação do conjunto de referência

Considere o conjunto P de soluções diversas melhoradas geradas no Passo 1. $RefSet_1$ é formado pelas b_1 melhores soluções de P . $RefSet_2$ é construído a partir das soluções de P que não pertencem a $RefSet_1$ e que maximizam a distância mínima em relação à $RefSet$.

3.2 Atualização do conjunto de referência

Existem dois aspectos que devem ser considerados na atualização de $RefSet$. O primeiro refere-se ao momento em que a atualização deve ser realizada. O segundo refere-se aos critérios de aceitação ou eliminação de uma solução de $RefSet$.

O primeiro aspecto considera dois tipos de atualização:

Atualização estática (S): Na atualização estática, o conjunto de referência não muda até que todas as combinações de soluções de $RefSet$ tenham sido realizadas. Exemplos de atualizações estáticas podem ser encontrados em Campos et al. (2001), Laguna e Martí (2004), Martí, Laguna e Campos (2002) e Campos, Laguna e Martí (2003).

Atualização dinâmica (D): Na atualização dinâmica, o conjunto de referência é atualizado sempre que uma nova solução, gerada por uma combinação, qualifica-se para entrar em $RefSet$. Nesse caso, algumas combinações do $RefSet$ original não são realizadas. Em geral, esse tipo de atualização acelera a convergência do método, pois soluções de alta qualidade substituem soluções de baixa qualidade. Exemplos de atualização dinâmica encontram-se em Martí, Lourenço e Laguna (2000), Jain e Meeran (2002), Laguna e Martí (2004), Yamashita, Armentano e Laguna (2006).

O segundo aspecto considera dois tipos de atualização: por qualidade e diversidade (QD) e apenas por qualidade (Q).

Atualização QD: Considere uma solução gerada pelo método de combinação de soluções. Se o custo da combinação resultante for menor do que a pior solução de $RefSet_1$, então a nova solução resultante substitui o pior elemento de $RefSet$. Se o custo for maior do que o pior elemento de $RefSet_1$, esta solução é inserida em $RefSet_2$ se aumentar a menor distância entre as soluções deste subconjunto. Nesse caso, a solução pertencente à $RefSet$ que possuir a menor distância em relação às demais é substituída pela nova solução.

Atualização Q: Se a nova solução gerada pelo método de combinação de soluções for melhor, em termos de qualidade, que o pior elemento de $RefSet$, então substitui-se o pior elemento de $RefSet$ pela nova solução.

4. Método de geração de subconjuntos

De acordo com o modelo de Glover (1998), quatro tipos de subconjuntos podem ser gerados:

- 1) **Subconjunto 1:** Todos os subconjuntos de 2 soluções.
- 2) **Subconjunto 2:** Subconjunto de 3 soluções, obtido ao incluir a melhor solução nos subconjuntos de 2 soluções, se essa solução não estiver presente neste subconjunto.
- 3) **Subconjunto 3:** Subconjunto de 4 soluções, obtido ao incluir a melhor solução nos subconjuntos de 3 soluções, se a melhor solução não estiver presente neste subconjunto.
- 4) **Subconjunto de 4:** Consistem os subconjuntos das i melhores soluções para $i = 5$ até b .

5. Método de combinação das soluções

O método de combinação utilizado no presente trabalho está detalhado no capítulo 6.

Um exemplo de aplicação da metaheurística SS para o problema da mochila encontra-se em anexo (ANEXO A). Neste exemplo, aplica-se o primeiro método para combinação das soluções.

3.2 APLICAÇÕES EM PROBLEMAS DE ROTEIRIZAÇÃO DE VEÍCULOS

A aplicação da metaheurística *scatter search* teve início na década de 90 e poucos estudos foram feitos em problemas de roteirização de veículos, nenhum deles em problemas de roteirização com entregas fracionadas (e suas extensões) e nem em problemas de roteirização com frota heterogênea. Este item tem como objetivo apresentar aplicações do algoritmo *scatter search* em problemas de roteirização de veículos.

Rego e Leão (2000) mostraram a importância da metaheurística *scatter search* para resolução de problemas de natureza combinatória, em especial problemas de roteirização de veículos. Os autores apresentaram um exemplo ilustrativo do procedimento *scatter search*, passo a passo, para uma instância do problema clássico de roteirização de veículos. Os autores classificaram o algoritmo como um método evolucionário para problemas de roteirização de veículos.

Corberán et al. (2002) implementaram a metaheurística *scatter search* para um problema de transporte escolar com múltiplos objetivos e um total de 58 ônibus escolares. Do ponto de vista de custo, busca-se minimizar o número de ônibus necessários para transportar os estudantes (ida e volta). Do ponto de vista de nível de serviço, busca-se minimizar o tempo de espera dos estudantes. O procedimento *scatter search* considera cada objetivo separadamente e busca um conjunto de soluções eficientes ao invés de uma única solução ótima. São utilizadas duas heurísticas construtivas para gerar soluções do conjunto de referência inicial. Na fase de melhoria, implementam-se procedimentos de trocas na mesma rota e inserção. O método de combinação gera novas soluções a partir de duas soluções originais (subconjuntos do tipo 1). Os autores utilizaram atualização estática e por qualidade.

Alegre, Laguna e Pacheco (2004) implementaram a metaheurística *scatter search* para um caso especial do problema de roteirização de veículos periódico (*Periodic Vehicle Routing Problem* – PVRP). O problema é baseado em uma fábrica de autopeças situada no norte da Espanha. A fábrica deve coletar peças (matérias-primas) de diversos fornecedores geograficamente dispersos e este procedimento é feito periodicamente. O problema consiste em determinar a programação das coletas para cada um dos fornecedores e estabelecer

roteiros diariamente, com o objetivo de minimizar o custo total de transporte. O horizonte de tempo considerado é de 90 dias. A fase de melhoria consiste na troca de clientes na mesma rota e trocas cruzadas. O método de atualização utilizado foi o estático e por qualidade. As soluções foram combinadas duas a duas. A distância entre duas soluções é calculada pelo número de arcos diferentes. Segundo os autores, a metaheurística é capaz de obter soluções de alta qualidade em um tempo computacional razoável.

Chu, Labadi e Prins (2006) implementaram a metaheurística SS para um problema de roteirização de veículos periódico com demanda em arcos (*Periodic Capacitated Arc Routing Problem* – PCARP). Resolve-se o CARP para cada período, com o objetivo de minimizar o tamanho da frota e o custo total de viagem no horizonte de planejamento. A solução inicial é obtida através de uma heurística gulosa. O algoritmo foi aplicado em dois conjuntos benchmark do PCARP, derivados do problema clássico CARP. Segundo os autores, através de SS foi possível melhorar substancialmente a solução inicial. SS também foi implementada para o caso de um único período.

Russell e Chiang (2006) utilizaram a metaheurística SS para resolução de um problema de roteirização de veículos com janelas de tempo (VRPTW). O conjunto de referência inicial é gerado a partir de duas heurísticas construtivas: heurística de construção paralela (Chiang e Russell, 1997) e heurística de inserção seqüencial de Solomon (1987). Os autores utilizaram duas versões da metaheurística busca tabu para a fase de melhoria. O tipo de atualização é por qualidade e diversidade. O método de combinação é baseado em duas fases: combinação dos arcos comuns das soluções e método de combinação das rotas. O algoritmo foi aplicado ao conjunto de problemas *benchmark* de Solomon (1987). Segundo os autores, o modelo garante soluções robustas, pois gera soluções de qualidade que são competitivas com os melhores resultados encontrados na literatura.

CAPÍTULO 4

DEFINIÇÃO DO PROBLEMA

Conforme dito anteriormente, o presente trabalho trata de um problema de transporte com decisões de seleção de veículos, consolidação das cargas e roteirização, a partir de um Centro de Distribuição (CD) que atenda as restrições de abastecimento das lojas, com o menor custo possível. O problema é classificado como um problema de roteirização de veículos com frota heterogênea, janelas de tempo e entregas fracionadas (HFVRPTWSD), onde os pontos de origem e destino são coincidentes. A definição completa e detalhada do problema está listada abaixo.

A partir de um único centro de distribuição, que distribui múltiplos produtos, são atendidos N clientes a cada dia. Os clientes são abastecidos através de uma frota de veículos heterogênea e ilimitada. Cada veículo tem uma capacidade C_v e um custo fixo e variável por viagem. Para um determinado dia, a demanda de cada cliente i é d_i . As lojas possuem restrições de janelas de tempo e tipo de veículo. Os veículos saem e retornam ao depósito.

Dada a demanda das lojas em um determinado dia, o objetivo do modelo é decidir como alocar os veículos às lojas (determinar os clientes de cada rota), determinar a quantidade a ser entregue em cada veículo para cada loja, qual o melhor roteiro no caso de rota fracionada e o tempo de início de atendimento do primeiro cliente da rota, de forma a minimizar o custo total de distribuição e o tempo total de viagem, respeitando todas as restrições do problema. As restrições são:

- Atender a demanda de todas as lojas;
- Respeitar as restrições de janelas de tempo de cada loja;
- Cada tipo de veículo tem uma capacidade C_v ;
- Restrições quanto ao tipo do veículo para algumas lojas.

As principais hipóteses e características do problema são:

- A demanda das lojas é determinística;
- Os produtos que compõem os pedidos são considerados iguais (considera-se apenas o volume ocupado e não o tipo de mercadoria);
- Todas as entregas são consideradas únicas e não periódicas, de modo que, a cada dia, os pedidos devem ser formulados novamente (horizonte de planejamento de um dia);
- A frota de veículos é heterogênea e ilimitada;
- Os custos de roteirização variam conforme o tipo do veículo;
- Os clientes podem receber mercadoria de mais de um veículo;
- Todos os veículos iniciam e terminam seu trajeto no depósito;
- O horário de início da jornada é flexível e varia de acordo com a necessidade de cada roteiro;
- O tempo de descarga é fixo em 1,5 horas para a maioria das lojas (grandes lojas);
- O centro de distribuição também abastece clientes finais (pequenos clientes) e, neste caso, o tempo de descarga é de apenas 15 minutos.
- O centro de distribuição trabalha 24 horas por dia.

4.1 CLASSIFICAÇÃO SEGUNDO DESROCHERS, LENSTRA E SAVELSBERGH (1990)

O problema estudado será classificado conforme Desrochers, Lenstra e Savelsbergh (1990).

CLASSIFICAÇÃO:

ENDERECOS

- número de depósitos: - um
- tipo de demanda: - localizada em nós
 - somente entregas
 - determinística
- restrições de programação dos endereços: - múltiplas janelas de tempo

- restrição de seleção dos endereços: - todos os endereços com demanda devem ser visitados.

VEÍCULOS

- número de veículos: - ilimitado
- restrição de capacidade: - frota heterogênea
- restrição de mercadoria: - não
- restrição de disponibilidade do veículo: - não
- restrição de tempo de duração da rota: - sim

CARACTERÍSTICAS DO PROBLEMA

- Tipo de rede: - euclidiana
 - não direcionada
- Tipo de estratégia de serviço: - a demanda não pode ser violada
 - entregas diretas ou fracionadas
 - roteirização envolve somente entrega
 - o veículo realiza uma rota por período
 - os veículos saem e retornam ao mesmo depósito
- Restrição endereço-endereço: - não há restrições de precedência
 - não há restrições entre depósito e cliente
 - não há restrições entre clientes
- Restrição endereço-veículo: - existem restrições entre clientes e veículos (tempo de descarga, recebimento no tipo de veículo)
 - não existem restrições entre depósito e veículos
- Restrição veículo-veículo: - não existem restrições entre veículos

FUNÇÃO OBJETIVO

- Minimizar soma de:
 - custos fixos dos veículos
 - custos variáveis (R\$/km)
 - tempo total de viagem

4.2 MODELAGEM MATEMÁTICA

Considere um conjunto de N clientes a serem atendidos. Em um determinado dia, a demanda d_i de cada cliente $i \in \{1, \dots, n\}$ deve ser atendida. O ponto 0 representa o depósito.

A cada cliente i estão associados:

- uma restrição de janela de tempo $[a_i, b_i]$, tal que $a_i \leq b_i$, que corresponde, respectivamente, ao horário inicial e final em que pode ser iniciado o atendimento;
- uma demanda d_i ;
- restrição quanto ao tipo de veículo;
- um tempo de atendimento s_i que representa o tempo de descarga dos veículos (tempo médio de todos os processos administrativos, burocráticos e de operação).

O número de tipos de veículos é representado por T . Há quatro tipos de veículos distintos: carreta, *truck*, toco e leve. Assume-se que o número de veículos de cada tipo é ilimitado. A cada veículo v estão associados:

- uma capacidade máxima C_v , que pode ser representada por uma restrição de peso e volume;
- um custo por viagem C_{viagem}^v , independente da distância total percorrida;
- um custo variável CV^v em função da distância (km), que corresponde aos custos de combustível, manutenção do veículo, pneus, lubrificantes, etc.

Um roteiro pode ser formado por uma única loja ou por várias lojas. A cada rota estão associados:

- Deslocamento do nó $i = 0$ (depósito) até um nó $j, j = 1, \dots, n$ (loja)
- Deslocamento entre lojas, no caso de carga fracionada (do nó $i, i = 1, \dots, n$ até o nó $j, j = 1, \dots, n$;
- Deslocamento do nó $i, i = 1, \dots, n$ (loja) até o depósito $j = 0$.

Cada deslocamento entre os nós i e j requer um tempo de viagem t_{ij} e uma distância percorrida d_{ij} . As matrizes de tempos de viagem e de distâncias são simétricas, ou seja, $t_{ij} = t_{ji}$ e $d_{ij} = d_{ji}$.

As variáveis de decisão do modelo são:

$x_{ij}^v = 1$, se j é atendido após i pelo veículo v ;

0, caso contrário.

T_i^v = hora de início de atendimento do cliente i pelo veículo v , $i = 1, \dots, n$ $v = 1, \dots, T$

y_i^v = fração de demanda do cliente i entregue pelo veículo v

O objetivo do modelo é minimizar o custo total de transporte (soma dos custos de viagem e dos custos variáveis em função da distância), de forma que a demanda de todos os clientes seja atendida e as demais restrições do problema sejam respeitadas. A formulação completa do problema está detalhada a seguir com base nos trabalhos de Dror e Trudeau (1990), Ho e Haugland (2004) e Golden et al. (1984).

A função objetivo pode ser escrita da seguinte forma:

$$\min \sum_{v=1}^T C_{viagem}^v \sum_{j=1}^n x_{0j}^v + \sum_{i=0}^n \sum_{j=0}^n \sum_{v=1}^T CV^v d_{ij} x_{ij}^v$$

As restrições do modelo são:

$$\sum_{j=1}^n x_{0j}^v = 1 \quad v = 1, \dots, T \quad (4.1)$$

A restrição (4.1) garante que cada veículo saia do depósito e chegue a um determinado cliente.

$$\sum_{i=0}^n x_{ip}^v - \sum_{j=0}^n x_{pj}^v = 0 \quad p = 0, \dots, n; \quad v = 1, \dots, T \quad (4.2)$$

A restrição (4.2) é a restrição de conservação dos fluxos de entrada e saída; garante que cada veículo saia de um determinado cliente e retorne ao depósito.

$$\sum_{v=1}^T y_i^v = 1, \quad i = 1, \dots, n \quad (4.3)$$

A restrição (4.3) garante que a demanda total de cada cliente será atendida.

$$\sum_{i=1}^n d_i y_i^v \leq C_v \quad v = 1, \dots, T \quad (4.4)$$

A restrição (4.4) garante que a capacidade de cada veículo não será excedida.

$$y_i^v \leq \sum_{j=0}^n x_{ji}^v \quad i = 1, \dots, n; \quad v = 1, \dots, T \quad (4.5)$$

A restrição (4.5) garante que a demanda de cada cliente será atendida somente se um determinado veículo passar por aquele ponto. Nota-se que, adicionando à restrição (4.5) o somatório de todos os veículos e, combinando-a com a equação (4.3), obtém-se a restrição $\sum_{v=1}^T \sum_{i=0}^n x_{ij}^v \geq 1 \quad j = 0, \dots, n$, que garante que cada vértice seja visitado pelo menos uma vez, por pelo menos por um veículo.

$$T_i^v + s_i + t_{ij} - M_{ij}(1 - x_{ij}^v) \leq T_j^k \quad i = 1, \dots, n; \quad j = 1, \dots, n; \quad v = 1, \dots, T \quad (4.6)$$

A equação (4.6) impõe o horário de início de atendimento mais cedo do cliente j em uma determinada rota e também garante que não haja formação de *subtours*. A constante M_{ij} é um número suficientemente grande, por exemplo, $M_{ij} = b_i + t_{ij} - a_j$.

$$a_i \leq T_i^v \leq b_i \quad i = 1, \dots, n \quad (4.7)$$

A restrição (4.7) garante que todos os clientes são atendidos dentro de sua janela de tempo.

Há também a restrição quanto ao tipo de veículo (4.8). Consideremos três conjuntos de veículos: $V_1 = \{\text{carreta, truck, toco e leve}\}$, $V_2 = \{\text{truck, toco e leve}\}$ e $V_3 = \{\text{carreta e truck}\}$. Algumas lojas i aceitam todos os tipos de veículos ($i \in V_1$). Muitas lojas i não aceitam o maior veículo (carreta) devido às restrições operacionais como tamanho das docas e falta de espaço para manobra ($i \in V_2$). Outras lojas i , fora do Estado de São Paulo, aceitam apenas carreta e *truck* ($i \in V_3$).

$$\begin{aligned} y_i^v &\geq 0 \quad i = 1, \dots, n; \quad v = 1, \dots, T \\ T_i^v &\geq 0 \quad i = 1, \dots, n \end{aligned} \tag{4.9}$$

A equação (4.9) garante que as variáveis de decisão y_i^v e T_i^v sejam positivas.

$$x_{ij}^v \in \{0, 1\} \quad i = 0, \dots, n; \quad j = 0, \dots, n; \quad v = 1, \dots, T \tag{4.10}$$

Finalmente a equação (4.10) garante que as variáveis de decisão x_{ij}^v sejam binárias.

4.3 COMPLEXIDADE DO PROBLEMA E ESTRATÉGIA DE SOLUÇÃO

Conforme descrito no item 2.1, o problema de roteirização de veículos com frota heterogênea, janelas de tempo e entregas fracionadas (*Heterogeneous Fleet Vehicle Routing Problem with time Windows and Split Deliveries* - HFVRPTWSD) é *NP*-completo, o que justifica o uso de heurísticas e metaheurísticas para sua resolução.

Desta forma, são implementadas heurísticas construtivas e a metaheurística *scatter search* para resolução do problema estudado. As soluções iniciais são obtidas através de quatro heurísticas construtivas: heurística de economias, adaptação da heurística de inserção seqüencial I1 de Solomon (1987), heurística de inserção seqüencial baseada nas idéias de Ho e Haugland (2004) e adaptação da heurística de inserção seqüencial de Dullaert et al. (2002).

CAPÍTULO 5

ESTUDO DE CASO

O algoritmo proposto é implementado para resolução de um problema real de abastecimento de lojas em um dos maiores grupos varejistas brasileiros. O grupo vem crescendo nos últimos anos através de uma contínua expansão da área de vendas, abrindo novas lojas e efetuando novas aquisições.

O estudo é efetuado a partir de um centro de distribuição que está localizado em Osasco e atende um total de 519 lojas em todo o Brasil. As lojas estão localizadas em todo o Estado de São Paulo e nas cidades de Brasília, Goiânia, Curitiba, Belo Horizonte, Rio de Janeiro, Campo Grande, Fortaleza, Teresina, Recife e João Pessoa. Os produtos são distribuídos através de uma frota de veículos heterogênea e ilimitada. O serviço de transporte é terceirizado.

Neste capítulo são discutidos detalhadamente como foram obtidos, tratados e inseridos todos os dados de entrada do modelo.

5.1 CENTRO DE DISTRIBUIÇÃO

O estudo de caso é implementado a partir de um centro de distribuição (CD), que compõe três CDs próximos entre si: mercearia seca, mercearia líquida e almoxarifado.

5.2 TIPOS DE VEÍCULOS

Todas as entregas de mercadorias são feitas através de uma frota terceirizada. Os modelos de veículos do CD utilizados nas operações da empresa, para o transporte de produtos de mercearia, estão listados a seguir:

- Carreta
- *Truck*

- Toco
- Leve

As capacidades em peso e volume para cada veículo, tanto para o interior de São Paulo (ou outros estados) quanto para a capital, estão na tabela abaixo.

Tabela 5.1 - Capacidade dos veículos

Tipo de veículo	Interior		Capital	
	Peso (t)	Volume (m³)	Peso (t)	Volume (m³)
Carreta	22.000	49	26.000	49
Truck	10.500	26	13.000	26
Toco	5.500	17	8.000	17
Leve	2.600	13	4.600	13

Cabe lembrar que há restrições quanto ao tipo de veículo para algumas lojas. Das lojas do Estado de São Paulo, a maioria delas (mais de 96%) não recebem carreta e aceitam os demais tipos de veículos. Já as lojas fora do Estado de São Paulo aceitam carreta e *truck*.

5.3 MODELO DE VALOR DE FRETE

Segundo Pestana e Gageiro (2000), um modelo de regressão linear simples analisa a relação entre duas variáveis de natureza quantitativa X e Y , de forma que esta relação possa ser representada através de uma reta, expressa pela seguinte equação:

$$\hat{Y}_i = a + bX_i + \varepsilon_i, \quad i = 1, 2, \dots, n \quad (5.1)$$

onde:

\hat{Y} = estimativa da variável dependente ou de resposta.

X = variável independente ou preditora.

a = constante ou valor médio de Y quando X é zero.

b = inclinação da reta ou coeficiente de regressão para a variável X_i : é a alteração (aumento ou diminuição) no valor médio de Y associado a um aumento unitário de X .

ε_i = variável aleatória residual que descreve os efeitos em Y_i não explicados por X_i .

$$\varepsilon_i = \text{valor real} - \text{valor previsto} = Y_i - \hat{Y}_i$$

Com base nos valores de fretes pagos às transportadoras, em função das lojas ou regiões e dos tipos de veículos, foi elaborado um modelo de regressão linear que determina o valor do frete do CD para as respectivas lojas, em função da distância total percorrida (ida e volta). A variável dependente utilizada no modelo é o valor do frete e a variável independente é a distância total percorrida.

O modelo determina os custos por viagem (constante) e os custos variáveis (inclinação da reta ou coeficiente de regressão) em função da distância total percorrida (ida e volta). A equação pode ser assim expressa:

$$\text{Valor do Frete}_i = C_{\text{viagem}} + CV * km_i + \varepsilon_i \quad (5.2)$$

Os resultados para cada tipo de veículo encontram-se na tabela abaixo.

Tabela 5.2 - Modelo de valor do frete

	Carreta	Truck	Toco	Leve
Custo por viagem (R\$/viagem)	166,199	117,593	80,942	58,403
Custo variável (R\$/km)	0,997	0,688	0,778	0,670

Cabe ressaltar que, para os quatro veículos, os modelos explicam 99% do valor do frete, ou seja, o coeficiente de correlação $R^2 = 0,99$.

No caso em que um veículo é designado para atender mais de uma loja, consideram-se os mesmos parâmetros de custos fixos e variáveis da tabela 5.2, e a distância total da rota (ida e volta).

A tabela 5.3 apresenta os valores dos fretes utilizados para a construção do modelo, em função de suas respectivas distâncias (apenas de ida).

Tabela 5.3 - Valores dos Fretes

KM	Valor do frete (R\$/viagem)			
	Carreta	Truck	Toco	Leve
15	173	124	110	82
30	190	138	117	90
43	235	157	141	113
45	222	164	140	108
58	244	180	154	119
60	244	180	154	119
69	295	197	173	142
70	258	172	146	124
71	260	173	154	125
72	298	199	174	143
81	305	203	183	146
84	353	235	205	170
85	308	206	185	148
89	370	247	215	178
90	329	220	187	158
96	380	253	215	182
98	387	258	220	186
103	340	227	204	163
108	391	261	227	188
110	364	242	218	175
113	358	239	215	172
129	465	310	273	223
133	429	286	243	206
138	433	289	260	208
147	475	316	269	228
159	573	382	325	275
163	511	341	297	245
165	518	345	293	248
189	682	454	386	327
190	613	409	348	294
201	649	433	368	312
204	659	439	373	316
216	779	519	441	374
221	731	487	414	351
227	712	475	403	342
264	828	552	469	397
302	947	631	568	455
346	1091	727	618	523
362	1202	801	681	577
383	1201	801	711	577
408	1057	763		
423	1405	937	796	674
434	1361	907	771	653
443	1242	821		
473	1421	933		
515	1710	1140	969	821
580	1688	1092		
586	1471	1001		
909	2309	1494		
1030	2224	1429		
1100	2384	1564		
1955	4122	2888		
2661	5409	3749		
2780	5456	3782		
2790	5734	3924		
3130	5846	4223		

5.4 DISTÂNCIA

Para se representar de maneira precisa a rede de lojas, de modo que possamos avaliar as distâncias percorridas nas rotas propostas, bem como o tempo necessário para percorrê-las, é necessário conhecer a localização de cada ponto.

De posse das coordenadas podemos calcular a distância entre todos os pares de pontos que compõem a rede. Caso estas coordenadas fossem a posição (x, y) dos pontos sobre um plano, poderíamos obter as distâncias calculando apenas o comprimento da reta que os une como sendo:

$$d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \quad (5.3)$$

No entanto, como as coordenadas estão em latitude e longitude, calcula-se o comprimento do arco que une os dois pontos sobre a superfície da Terra, conhecida como distância geodésica. As latitudes e longitudes foram obtidas a partir dos seus respectivos endereços com o auxílio de um *site* de localização (www.kretta.com.br). No caso de municípios nos quais as informações de latitude e longitude não estavam disponíveis, foram considerados os centróides da cidade. O cálculo da distância geodésica, de acordo com Ballou (2001), encontra-se em anexo (ANEXO B).

No entanto, a distância geodésica na prática irá depender das vias terrestres disponíveis entre os pontos. Para corrigir esta distorção entre o valor teórico e o valor prático das distâncias foi feita uma análise com base nos dados históricos da distância percorrida e do resultado da equação anterior. Com isso, obteve-se o fator de correção da distância geodésica, que foi de 1,27 para os trechos urbanos e 1,25 para os demais.

5.5 VELOCIDADES E TEMPO DE DESCARGA

Com o intuito de determinar as velocidades de percurso, foi feita uma análise com base nos dados históricos da empresa. Comparou-se a distância real percorrida entre dois pontos e o tempo total de percurso (sem incluir as paradas para carga e descarga do veículo), obtendo-se assim, a velocidade de percurso para cada trecho. Os dados completos, que incluem velocidade média de percurso e tempo de descarga, encontram-se na tabela abaixo.

Tabela 5.4 - Tempo de descarga e velocidade média

Tempo de descarga	1,5 horas	Grandes lojas
	15 minutos	Consumidores finais
	20 km/h	Urbano
Velocidade média de percurso	30 km/h	Média
	45 km/h	Semi
	58 km/h	Longo curso

5.6 DEMANDA DAS LOJAS

A demanda das lojas é dada em peso (t), volume (m^3) e número de paletes. A capacidade do veículo não pode exceder nenhuma destas medidas. A tabela a seguir apresenta a demanda de algumas lojas em um determinado dia.

Tabela 5.5 - Demanda das lojas

Loja	Peso	Volume
1	9.640	20,3
6	7.512	19,6
7	3.191	9,2
9	6.171	13
11	1.846	4,6
16	6.436	14,2
20	6.157	16,1
21	6.916	17,4
25	2.333	5,4
28	3.051	7,7
35	11.107	22,8
37	8.814	17,8
39	3.672	7,2
44	4.746	11,5
48	1.414	4,3
50	8.157	17,6
52	5.429	10

5.7 JANELAS DE TEMPO DAS LOJAS

A tabela abaixo mostra as janelas de tempo de algumas lojas.

Tabela 5.6 - Janelas de recebimento

Loja	Início do Recebimento (a_i)	Final do Recebimento (b_i)
1	12:00	15:00
6	18:00	20:00
7	11:00	15:00
8	06:00	08:00
9	19:00	21:30
11	13:00	18:00
12	08:00	14:00
16	19:00	22:00
18	06:00	22:00
19	06:00	07:00
20	20:00	22:00
21	21:00	23:00
25	06:00	08:00
26	11:00	16:00
28	21:00	03:00
35	06:00	10:00

CAPÍTULO 6

RESOLUÇÃO DO MODELO

Este capítulo tem como objetivo apresentar detalhadamente os algoritmos implementados para a resolução do problema estudado. O presente trabalho busca contribuir no desenvolvimento de novos métodos de solução em VRP, através da implementação da metaheurística *scatter search* (SS) para o problema de roteirização de veículos com frota heterogênea, janelas de tempo e entregas fracionadas (HFVRPTWSD). O conjunto de referência inicial do algoritmo *scatter search* é formado a partir das soluções obtidas das heurísticas construtivas. São implementadas quatro heurísticas construtivas, uma delas baseada na heurística de economias de Clarke e Wright (1964) e as heurísticas restantes são algoritmos de inserção seqüencial. As heurísticas construtivas têm como objetivo gerar soluções iniciais factíveis. A metaheurística SS busca explorar novos espaços de soluções, a fim de melhorar a solução inicial obtida.

As informações foram armazenadas no banco de dados SQL e os algoritmos foram implementados na linguagem de programação Delphi 7.0., em uma máquina AMD Athlon com processador de 1.99 GHz e 1 Gb de memória RAM.

Os itens 6.1, 6.2, 6.3 e 6.4 descrevem, detalhadamente, cada uma das heurísticas construtivas implementadas para o problema estudado. No item 6.5 estão descritos todos os passos da metaheurística *scatter search*. Este capítulo também descreve as adaptações necessárias em relação ao algoritmo proposto para resolução de três conjuntos de problemas encontrados na literatura: Solomon (1987), Ho e Haugland (2004) e Liu e Shen (1999).

6.1 HEURÍSTICA DE ECONOMIAS (VERSÃO PARALELA)

Neste item é descrita a implementação detalhada da versão paralela da heurística de economias de Clarke e Wright (1964) adaptada.

6.1.1 FROTA HETEROGÊNEA

A heurística básica de Clarke e Wright adota frota de veículos homogênea. Neste trabalho, a frota de veículos é heterogênea, o que aumenta a complexidade do problema. No caso de problemas com frota heterogênea é importante determinar se é melhor utilizar mais veículos de menor capacidade, percorrendo uma distância total menor ou menos veículos de maior capacidade, percorrendo uma distância total maior, de forma a otimizar o custo total.

As seguintes adaptações foram feitas à heurística original de economia paralela de Clarke e Wright:

- Assim como no algoritmo original, inicialmente são criadas rotas individuais para atender cada uma das lojas;
- A cada rota é atribuído o menor caminhão capaz de atender a demanda da respectiva loja, conforme proposto por Golden et al. (1984);
- Quando duas lojas são avaliadas para se juntar em uma mesma rota, caso a demanda das duas combinadas ultrapasse a capacidade do veículo atual, o menor veículo capaz de atender essa demanda é designado para a rota.

Seguindo estes passos durante a execução da heurística de Clarke e Wright, garantimos que, no final do processo, apenas rotas que respeitem a capacidade dos veículos sejam formadas e que os menores veículos possíveis serão alocados para cada rota, de forma a minimizar o custo total de transporte.

6.1.2 DEMANDA MAIOR QUE A CAPACIDADE DOS VEÍCULOS

Como foi observado na revisão da literatura, uma das hipóteses básicas da heurística de Clarke e Wright é que a demanda de cada uma das lojas não pode exceder a capacidade dos veículos, pois neste caso a mesma não poderia ser atendida em apenas uma parada. No entanto, conforme mencionado anteriormente, a maioria das lojas possui demanda superior à

comportada pelos veículos.

Neste sentido, foi implementado um algoritmo para suprir a demanda de cada loja com caminhões lotados visitando exclusivamente aquela loja, até que a demanda remanescente seja menor do que a capacidade do maior veículo capaz de abrigá-la. Assim, garantimos que a demanda de cada loja que será roteirizada pela heurística é menor do que a capacidade dos caminhões, permitindo que os métodos pesquisados sejam utilizados sem maiores problemas. Na prática, este método é bastante intuitivo, pois, enviar veículos grandes com carga completa (*truckload*), é a forma de transporte que minimiza o custo unitário por unidade entregue.

O algoritmo proposto segue os seguintes passos:

Figura 6.1 Algoritmo para atendimento das lojas com demanda superior à capacidade dos veículos

- Para cada loja:

Enquanto a demanda remanescente for maior do que a capacidade do maior veículo capaz de atender a loja **faça**

Envie um caminhão deste tipo com carga completa

Recalcule a demanda remanescente

Fim enquanto

6.1.3 CÁLCULO DAS ECONOMIAS

Uma das deficiências da heurística Clarke e Wright é que ela minimiza apenas a distância total percorrida, não considerando os custos dos veículos. A economia corresponde à distância economizada ao juntar duas lojas numa mesma rota. Porém, neste trabalho, o objetivo do modelo é minimizar o custo total de transporte. Conforme descrito no capítulo 5, o custo para atender uma única loja corresponde ao custo fixo do veículo por viagem, mais um custo variável multiplicado pela distância de ida e volta do Centro de Distribuição ($i = 0$) à loja, resultando em um custo total C_i para atender a loja i ($i = 1, \dots, n$). No caso em que o veículo visita n lojas i ($i = 1, \dots, n$), o custo total da rota corresponde ao custo fixo do veículo por viagem, mais um custo variável do veículo multiplicado pela distância total percorrida na rota que corresponde à soma das distâncias do Centro de distribuição à loja $i = 1$, entre as lojas, e da loja $i = n$ até o Centro de Distribuição.

Considerando dois clientes i e j com entregas separadas, o custo total de transporte do cliente i é:

$$C_i = CF + CV * 2d_{0,i} \quad (6.1)$$

Já o custo total de transporte do cliente j , considerando o mesmo veículo, é:

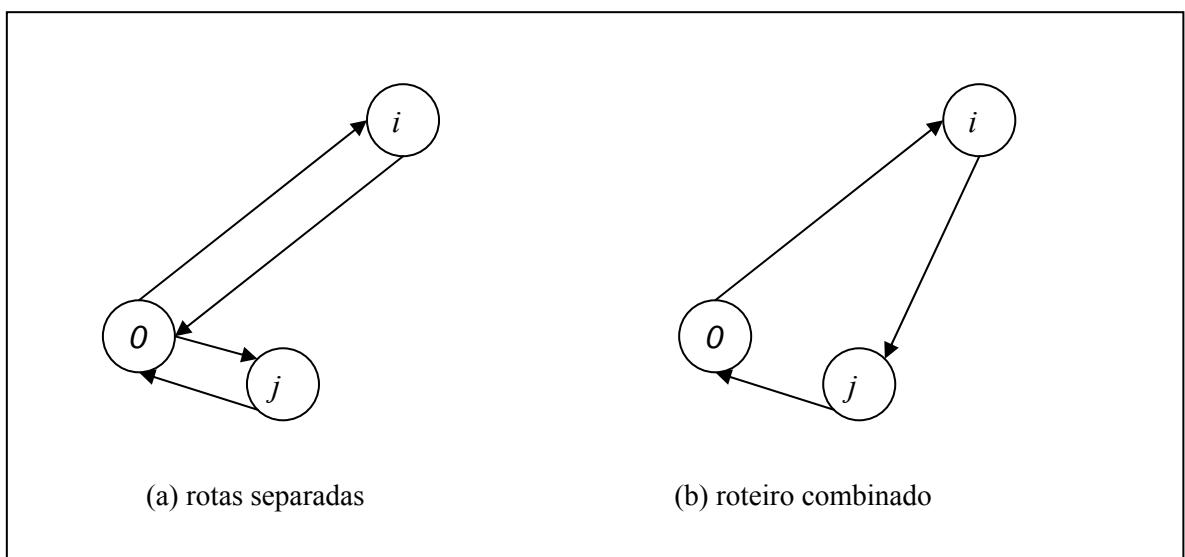
$$C_j = CF + CV * 2d_{0,j} \quad (6.2)$$

Portanto, o custo total de entrega dos clientes i e j é:

$$C_T = 2CF + CV * (2d_{0,i} + 2d_{0,j}) \quad (6.3)$$

Uma possibilidade de melhoria desse esquema seria juntar os dois clientes i e j no mesmo roteiro, conforme mostra a figura 6.2.

Figura 6.2 - Integrando dois clientes em um roteiro compartilhado



Considerando que os clientes i e j passarão a fazer parte do mesmo roteiro de entrega, e considerando o mesmo veículo utilizado para entregas separadas, o custo total da rota combinada C_T' pode ser definido como:

$$C_T' = CF + CV(d_{0,i} + d_{i,j} + d_{j,0}) \quad (6.4)$$

Portanto, a economia obtida pela junção das lojas i e j no mesmo roteiro seria:

$$S_{ij} = C_T - C_T' = CF + CV(d_{0,i} + d_{0,j} - d_{ij}) \quad (6.5)$$

Num caso genérico, a **função economia** pode ser definida como:

$$S_{ij} = CF_i + CF_j - CF_{i,j} + CV_i(2d_{0,i}) + CV_j(2d_{0,j}) - CV_{i,j}(d_{0,i} + d_{ij} + d_{j,0}) \quad (6.6)$$

onde:

CF_i = Custo fixo do menor veículo capaz de atender a demanda do cliente i

CF_j = Custo fixo do menor veículo capaz de atender a demanda do cliente j

CF_{ij} = Custo fixo do menor veículo capaz de atender as demandas dos clientes i e j

CV_i = Custo variável do menor veículo capaz de atender a demanda do cliente i

CV_j = Custo variável do menor veículo capaz de atender a demanda do cliente j

CV_{ij} = Custo variável do menor veículo capaz de atender as demandas dos clientes i e j

Como a frota de veículos é heterogênea, a cada junção de rotas a lista de economias é recalculada em função do novo tipo de veículo. A cada iteração não é necessário recalcular todas as economias da lista, e sim, apenas as economias ligadas à última junção de rotas.

6.1.4 RESTRIÇÕES DE JANELAS DE TEMPO, CAPACIDADE E TIPO DE VEÍCULO

A complexidade do problema aumenta quando a frota de veículos utilizada é heterogênea. Algumas lojas possuem restrições quanto ao tipo de caminhão devido às suas dimensões não serem compatíveis com as docas de recebimento disponíveis ou não haver espaço de manobra, entre outros motivos. Além disso, a maioria das lojas tem um horário pré-determinado para recebimento da mercadoria.

Para a junção de lojas em uma mesma rota, além das restrições de capacidade dos veículos, as restrições de janelas de tempo e tipo de veículo não podem ser violadas.

No caso da empresa estudada há três categorias de lojas em relação ao recebimento de veículos:

- **Categoria 1:** aceitam todos os tipos de veículos (carreta, *truck*, toco e leve), e correspondem a apenas 19 das 519 lojas (menos de 4%), todas localizadas no Estado de São Paulo;
- **Categoria 2:** aceitam todos os veículos com exceção da carreta (*truck*, toco e leve), e correspondem a 385 das 519 lojas (74%), todas elas localizadas no Estado de São Paulo;
- **Categoria 3:** aceitam apenas carreta e *truck* e correspondem a 115 das 519 lojas (22%), todas elas localizadas fora do Estado de São Paulo (Salvador, Brasília, Goiânia, Curitiba, Belo Horizonte, Rio de Janeiro, Campo Grande, Fortaleza, Teresina, Recife e João Pessoa).

Foram necessárias algumas adaptações na heurística original para que todas as restrições pudessem ser respeitadas, tanto no atendimento inicial da demanda, como já foi explicado no início desta seção, quanto na avaliação da viabilidade na junção de duas lojas numa rota. Para que a junção de duas lojas possa ser efetuada, as restrições de capacidade do veículo, janelas de tempo e tipo de veículo devem ser respeitadas. O tipo de veículo designado é o menor capaz de atender a demanda da rota.

O algoritmo completo de economias pode ser formulado a seguir:

Figura 6.3 - Algoritmo completo de economias (versão paralela)

Passo 1

- Para cada loja:

Enquanto a demanda remanescente for maior do que a capacidade do maior veículo capaz de atender a loja **faça**

Envie um caminhão deste tipo com carga completa

Recalcule a demanda remanescente

Fim enquanto

Continuação da Figura 6.3 - Algoritmo completo de economias (versão paralela)

Passo 2

- Assim como no algoritmo original, inicialmente são criadas rotas individuais para atender a demanda remanescente de cada uma das lojas

Para cada rota é atribuído o **menor caminhão** capaz de atender a demanda da respectiva loja

Passo 3

- Combinar todas as lojas duas a duas e calcular a economia para cada combinação de acordo com a **função economia**.

Passo 4

- Ordenar as economias em ordem decrescente

Passo 5

- Iniciar com a combinação de dois nós que apresentou a maior economia. Posteriormente, na análise de outras situações, vai-se descendo na lista de combinações, sempre obedecendo à seqüência decrescente de economias.

Passo 6 (Construção paralela)

- Para um par de pontos (i, j) tirado da seqüência de combinações, verificar se os dois pontos já fazem parte de um roteiro iniciado:
 - a) se i e j não foram incluídos em nenhum dos roteiros já iniciados, criar, então, um novo roteiro com esses dois pontos, desde que satisfaça as restrições de capacidade, tipo de veículo e janela de tempo (orientação da rota) e definir o tipo de veículo (menor veículo capaz de atender a demanda da rota).
 - b) se i já pertence a um roteiro iniciado, verificar se esse ponto é o primeiro ou último desse roteiro (não contando o CD) e se o cliente j não está inserido em nenhum roteiro. Se a resposta for positiva, acrescentar o par de pontos (i, j) na extremidade apropriada, desde que satisfaça as restrições de capacidade, tipo de veículo e janela de tempo. O tipo de veículo designado é o menor capaz de atender a demanda da rota. Fazer a mesma análise com o ponto j . Caso i ou j já pertença a um roteiro iniciado, mas não seja um ponto extremo, passar para a próxima economia da lista. Se nenhum dos dois pontos satisfaz essa condição **separadamente**, de que apenas um ponto é extremo de um dos roteiros, passar para o item (c).

Continuação da Figura 6.3 - Algoritmo completo de economias (versão paralela)

Continuação do Passo 6 (Construção paralela)

c) se ambos os pontos i e j fazem parte, cada um deles, de roteiros iniciados, mas diferentes, verificar se ambos são extremos dos respectivos roteiros. Se a resposta for positiva, fundir os dois roteiros em um só, desde que satisfaça as restrições de capacidade, tipo de veículo e janela de tempo. O tipo de veículo designado é o menor capaz de atender a demanda da nova rota. Caso contrário, passar para o próximo elemento da lista.

Passo 7

A cada junção de rotas a lista de economias deve ser recalculada e reordenada em função do novo tipo de veículo. A cada iteração recalcular apenas as economias ligadas à última junção de rotas.

Passo 8

O processo termina quando todos os pontos (clientes) estiverem incluídos em um roteiro.

6.2 ADAPTAÇÃO DA HEURÍSTICA DE INSERÇÃO SEQUENCIAL I1 DE SOLOMON (AI1)

A heurística de inserção seqüencial I1 de Solomon (1987) tem como características frota homogênea, demanda de clientes menor que a capacidade dos veículos e clientes sendo atendidos por apenas um veículo, e o objetivo é minimizar o acréscimo de distância e tempo causados pela inserção de um cliente.

Este trabalho implementa uma adaptação da heurística de Solomon. O problema tem como características frota de veículos heterogênea, demanda de clientes maior ou menor que a capacidade dos veículos e clientes que podem ser atendidos por mais de um veículo, e o objetivo é minimizar o acréscimo de custo de transporte e tempo causado pela inserção de um cliente.

O passo 1 implementado na heurística de economias também é implementado neste modelo, de modo que a demanda remanescente de uma determinada loja seja menor que a capacidade do maior veículo capaz de atendê-la. A partir daí, aplica-se a heurística de inserção seqüencial.

O critério de início de rota utilizado é o cliente mais distante. A etapa seguinte é dividida em dois passos e corresponde à inserção de um cliente na rota.

Primeiramente, determina-se a melhor posição de inserção para cada cliente u . Calcula-se para cada cliente u ainda não alocado a nenhuma rota, a melhor posição viável de inserção, que é aquela que minimiza o acréscimo de custo de transporte e tempo causados após inserção.

O segundo passo é determinar o cliente u a ser inserido na rota, que é aquele que tem o menor acréscimo de custo e tempo (passo A). Os clientes são inseridos na rota enquanto houver capacidade no veículo e viabilidade de inserção dos clientes. Caso contrário, inicia-se uma nova rota e o procedimento é repetido enquanto houver cliente a ser alocado.

O algoritmo completo de inserção seqüencial pode ser escrito da seguinte forma:

Figura 6.4 - Descrição completa do algoritmo AI1

Passo 1

- Para cada loja:

Enquanto a demanda remanescente for maior do que a capacidade do maior veículo capaz de atender a loja **faça**

Envie um caminhão deste tipo com carga completa

Recalcule a demanda remanescente

Fim enquanto

Passo 2: Baseado na demanda remanescente

Programa Principal

Enquanto tiver cliente ainda não alocado **faça**

Executar Rotina Início de Rota

Enquanto for possível inserir um cliente na rota em construção **faça**

Executa Rotina Verificação se o cliente u ainda não alocado é viável e escolha do novo veículo

Executa Rotina Escolha da melhor posição para cada cliente u viável

Executa Rotina Escolha do melhor cliente u^* a ser inserido

Fim enquanto

Fim enquanto

Continuação da Figura 6.4 - Descrição completa do algoritmo AII

Início Rotina Início de Rota

Escolhe-se o cliente i mais distante ainda não alocado

Tipo de veículo = menor veículo viável capaz de atender a demanda do cliente i

Fim Rotina Início de Rota

Início Rotina Verificação se o cliente u ainda não alocado é viável e escolha do novo veículo

Tipo de veículo = menor veículo viável (em termos de capacidade e tipo de veículo) capaz de atender a demanda da rota

Fim Rotina Verificação se o cliente u ainda não alocado é viável e escolha novo veículo

Início Rotina Escolha da melhor posição para cada cliente u viável

Enquanto houver um cliente u viável ainda não alocado **faça**

Passo A: Calcular c_1 para cada posição p viável ($1 \leq p \leq m$)

$$\begin{aligned} c_1(i, u, j) &= \alpha_1 c_{11}(i, u, j) + \alpha_2 c_{12}(i, u, j) & \alpha_1 + \alpha_2 &= 1 \\ \alpha_1 &\geq 0, \alpha_2 \geq 0 \\ u &\geq 0 \end{aligned}$$

$c_{11}(i, u, j)$ = acréscimo no custo após a inclusão cliente $u = c_{iu} + c_{uj} - c_{ij}$ (mesmo veículo)

$c_{12}(i, u, j)$ = acréscimo de tempo após a inclusão do cliente $u = b_{j/u} - b_j$

onde:

b_j = início de serviço do cliente j

$b_{j/u}$ = início de serviço do cliente após inclusão do cliente j

Passo B: Calcular a melhor posição viável (em termos de janelas de tempo) para cada cliente u , que é aquela que minimiza o acréscimo de custo e tempo

$$c_1(i(u), u, j(u)) = \min \{c_1(i_{p-1}, u, i_p)\} \quad p = 1, \dots, m$$

Fim Rotina Escolha da melhor posição para cada cliente u viável

Início Rotina Escolha do melhor cliente u^* a ser inserido

É aquele que minimiza $c_1(i, u, j)$

Fim Rotina Escolha do melhor cliente u^* a ser inserido

6.3 ADAPTAÇÃO DA HEURÍSTICA DE INSERÇÃO SEQUENCIAL DE HO E HAUGLAND (AHH)

Esta heurística foi baseada nas idéias de Ho e Haugland (2004) e é uma extensão da heurística de inserção seqüencial descrita no item 6.2.

O critério de início utilizado é o cliente mais distante ainda não alocado. O critério de inserção é semelhante à heurística anterior e é aquele que minimiza o acréscimo de custo e tempo causados pela inserção de um cliente. Porém, ao invés de considerar apenas a demanda remanescente, considera-se a demanda total de cada loja (elimina-se o passo 1).

A heurística inicia-se com o menor veículo capaz de atender a demanda de um determinado cliente i . Caso a demanda do cliente i seja maior do que a capacidade do maior veículo, envia-se um veículo com carga completa e a demanda remanescente é adicionada ao menor veículo capaz de atendê-la. A próxima etapa é a inserção de um novo cliente j na rota. Caso a demanda total do cliente j , somada com a demanda do cliente i já existente, exceda a capacidade do maior veículo, a demanda do cliente j é adicionada, enquanto houver capacidade, e a demanda remanescente é adicionada a um novo veículo (menor veículo capaz de atender a demanda remanescente). Este processo continua até que todos os clientes pertençam a um roteiro.

6.4 ADAPTAÇÃO DA HEURÍSTICA DE INSERÇÃO SEQUENCIAL DE DULLAERT ET AL. (AHD)

Dullaert et al. (2002) adaptaram a heurística de inserção seqüencial de Solomon (1987) para um problema de dimensionamento e roteirização de uma frota heterogênea de veículos com janelas de tempo (FSMVRPTW).

A AHD implementada no presente trabalho é similar à heurística de Dullaert, acrescentando o passo 1 da heurística de economias e da AI1, de modo que a demanda remanescente de cada cliente seja menor que a capacidade do maior veículo capaz de atendê-lo.

Esta heurística difere da heurística AI1 apenas no cálculo da Rotina Escolha da melhor posição para cada cliente u viável e da Rotina Escolha do melhor cliente u^* a ser inserido.

O critério de inserção $c_1(i, u, j)$ leva em conta o acréscimo de distância $c_{11}(i, u, j)$ e tempo $c_{12}(i, u, j)$ causados pela inserção de um cliente u , além de um terceiro componente $c_{13}(i, u, j)$ que considera um possível acréscimo no custo fixo do veículo. Porém, ao contrário da heurística I1 proposta por Solomon (1987), não há restrição de que a soma dos pesos α_i seja 1.

O terceiro componente $c_{13}(i, u, j)$ do custo de inserção $c_1(i, u, j)$ é uma adaptação do conceito de economia de Golden et al. (1984). Dullaert et al. (2002) implementaram 3 variações do conceito de economia: o conceito de economia combinada adaptada (*Adapted Combined Savings – ACS*), o conceito de economia de oportunidade otimista adaptada (*Adapted Optimistic Opportunity Savings – AOOS*) e o conceito de economia de oportunidade realista adaptada (*Adapted Realistic Opportunity Savings – AROS*). O presente trabalho implementa apenas o conceito de economia de oportunidade otimista adaptada (AOOS), em função dos resultados obtidos por Dullaert et al.

Segundo os autores, a carga atual de um veículo e sua capacidade máxima são denominadas Q e \bar{Q} , respectivamente. A nova carga do veículo e a possível nova capacidade depois da inserção de um novo cliente são representadas por Q^{new} e \bar{Q}^{new} , respectivamente.

O conceito de economia combinada adaptada (*Adapted Combined Savings – ACS*) é definido como a diferença entre o custo fixo do veículo capaz de transportar a carga da rota depois e antes da inserção do cliente u , $ACS = F(Q^{new}) - F(Q)$, onde $F(z)$ é o custo fixo do menor veículo capaz de entregar a demanda z .

O conceito de economia de oportunidade otimista adaptada (*Adapted Optimistic Opportunity Savings – AOOS*) é uma extensão de ACS subtraindo $F(\bar{Q}^{new} - Q^{new})$ que pode ser definido como o custo fixo do menor veículo capaz de transportar a carga não utilizada do novo veículo. Temos, portanto, que $AOOS = (F(Q^{new}) - F(Q)) - F(\bar{Q}^{new} - Q^{new})$.

Através da Rotina Escolha do melhor cliente u^* a ser inserido, seleciona-se o cliente u factível que maximiza o critério de seleção $c_2(i, u, j)$. O cliente selecionado u^* é então inserido na rota entre os clientes i e j .

O algoritmo completo está descrito a seguir.

Figura 6.5 - Descrição completa do algoritmo AHD

Passo 1

- Para cada loja:

Enquanto a demanda remanescente for maior do que a capacidade do maior veículo capaz de atender a loja **faça**

 Envie um caminhão deste tipo com carga completa

 Recalcule a demanda remanescente

Fim enquanto

Passo 2: Baseado na demanda remanescente

Programa Principal

Enquanto tiver cliente ainda não alocado **faça**

 Executar Rotina Início de Rota

Enquanto for possível inserir um cliente na rota em construção **faça**

 Executa Rotina Verificação se o cliente u ainda não alocado é viável e escolha do novo veículo

 Executa Rotina Escolha da melhor posição para cada cliente u viável

 Executa Rotina Escolha do melhor cliente u^* a ser inserido

Fim enquanto

Fim enquanto

Início Rotina Início de Rota

 Escolhe-se o cliente i mais distante ainda não alocado

 Tipo de veículo = menor veículo viável capaz de atender a demanda do cliente i

Fim Rotina Início de Rota

Continuação da Figura 6.5 - Descrição completa do algoritmo AHD

Início Rotina Verificação se o cliente u ainda não alocado é viável e escolha do novo veículo

Tipo de veículo = menor veículo viável (em termos de capacidade e tipo de veículo) capaz de atender a demanda da rota

Fim Rotina Verificação se o cliente u ainda não alocado é viável e escolha novo veículo

Início Rotina Escolha da melhor posição para cada cliente u viável

Enquanto houver um cliente u viável ainda não alocado **faça**

Passo 1: Calcular c_1 para cada posição p viável ($1 \leq p \leq m$)

$$c_1(i, u, j) = \alpha_1 c_{11}(i, u, j) + \alpha_2 c_{12}(i, u, j) + \alpha_3 c_{13}(i, u, j)$$

onde:

$$c_{11}(i, u, j) = d_{iu} + d_{uj} - ud_{ij}, \quad u \geq 0$$

$$c_{12}(i, u, j) = b_{j/u} - b_j$$

$$c_{13}(i, u, j) = \text{AOOS}$$

Passo 2: Calcular a melhor posição para cada cliente u , que é aquela que minimiza o acréscimo de distância, tempo e custo fixo do veículo

$$c_1(i(u), u, j(u)) = \min_p [c_1(i_{p-1}, u, i_p)], \quad p = 1, \dots, m$$

Fim Rotina Escolha da melhor posição para cada cliente u viável

Início Rotina Escolha do melhor cliente u^* a ser inserido

$$c_2(i, u^*, j) = \max_u [c_2(i, u, j)] \quad u \text{ não selecionado e factível}$$

$$c_2(i, u, j) = \lambda(d_{0u} + t_{0u}) + s_u + F(q_u) - c_1(i, u, j), \quad \lambda \geq 0$$

onde s_u é o tempo de serviço do cliente u

$F(q_u)$ é o custo fixo do menor veículo capaz de entregar a quantidade q_u

Fim Rotina Escolha do melhor cliente u^* a ser inserido

6.5 SCATTER SEARCH

Neste item é detalhado cada um dos passos da implementação da metaheurística *scatter search*.

1. Método de geração de soluções diversas

O conjunto de P soluções diversas com $Psize$ elementos é gerado através de um método que incorpora elementos aleatórios nas heurísticas analisadas, denominado método GRASP CONSTRUTIVO.

O procedimento GRASP (*Greedy Randomized Adaptive Search Procedure*) é uma técnica desenvolvida por Feo e Resende (1989) que consiste em duas fases: fase de construção e fase de melhoria. A fase de construção gera diversas soluções de boa qualidade através de um componente aleatório. A segunda fase consiste na implementação de movimentos de melhoria, com o objetivo de melhorar as soluções iniciais obtidas. O método GRASP CONSTRUTIVO é baseado na primeira fase do procedimento GRASP.

O objetivo deste método é, portanto, incorporar elementos aleatórios nas heurísticas construtivas através de um parâmetro α , gerando novas soluções.

No caso da heurística de economias, ao invés de escolher o primeiro elemento da lista de economias, o método GRASP CONSTRUTIVO sorteia um elemento, entre os k primeiros elementos da lista. O número de elementos k a serem considerados é a porcentagem α do número total de elementos.

No caso das heurísticas de inserção seqüencial, o método GRASP CONSTRUTIVO foi aplicado tanto no critério de início da rota, quanto no critério de inserção. Consideremos um exemplo em que o critério de início é o cliente mais distante. Desta forma, o método GRASP CONSTRUTIVO sorteia um elemento, entre os k elementos mais distantes, para inicializar a rota. Já no critério de inserção, insere-se um cliente entre os k clientes mais próximos. O critério de proximidade é calculado em função do acréscimo de custo de transporte e tempo causados pela inserção de um cliente na rota.

Neste trabalho são consideradas 2 variações do método GRASP CONSTRUTIVO, nomeadas GRACO1 e GRACO2. Em GRACO1 foi considerado $\alpha = \frac{i}{n} \times 100$, quando a i -ésima solução é gerada, em qualquer iteração (passagem pelo passo de diversificação). Por

exemplo, na primeira iteração escolhe-se o primeiro elemento da lista (solução semelhante à heurística construtiva). Já na segunda iteração, escolhe-se um elemento, aleatoriamente, entre os dois primeiros elementos da lista, e assim por diante, até que sejam geradas P soluções.

Em GRACO2, considerou-se na primeira iteração $\alpha = 5\%$, na segunda $\alpha = 10\%$ e assim por diante, até $\alpha = 45\%$ (nona iteração). Nas demais iterações considerou-se $\alpha = 50\%$. Por exemplo, na primeira iteração escolhe-se um elemento, aleatoriamente, entre os 5% primeiros elementos da lista. Na segunda iteração escolhe-se um elemento, aleatoriamente, entre os 10% primeiros elementos da lista, e assim por diante, até $\alpha = 45\%$. A partir da décima iteração, escolhe-se um elemento, aleatoriamente, entre os 50% primeiros elementos da lista, até que sejam geradas P soluções.

O método GRACO1 é aplicado para as três primeiras iterações da metaheurística *scatter search*. A partir da quarta iteração, utiliza-se o método de diversificação GRACO2.

Com o objetivo de selecionar as soluções iniciais do conjunto de referência, verificou-se, primeiramente, o desempenho de cada uma das heurísticas construtivas. O pior resultado foi obtido pela heurística de economias. A AHD (adaptação da heurística de inserção seqüencial de Dullaert et al.) gerou resultados um pouco superiores à AI1 (adaptação da heurística de inserção I1 de Solomon). Já a AHH (heurística de inserção seqüencial de Ho e Haugland adaptada) obteve desempenho inferior às heurísticas AI1 e AHD em termos de custo total, porém, superior em relação ao número de veículos necessários.

Desta forma, foram gerados 3 conjuntos de soluções diversas com base nas heurísticas construtivas AHD e AHH:

- a) Caso 1: 100% das soluções diversas são geradas a partir da heurística de inserção AHD.
- b) Caso 2: 100% das soluções diversas são geradas a partir da heurística de inserção AHH.
- c) Caso 3: 50% das soluções diversas são geradas a partir da heurística de inserção AHD e as 50% restantes a partir da heurística de inserção AHH.

A metaheurística *scatter search* é aplicada para cada um dos 3 conjuntos gerados, com o objetivo de identificar o melhor conjunto de soluções diversas. Os resultados detalhados das heurísticas construtivas, além da identificação do melhor conjunto de soluções diversas, estão no próximo capítulo.

2. Método de Melhoria

Para cada uma das P soluções obtidas a partir do método de geração de soluções diversas, aplica-se o método de melhoria que consiste em 5 fases, na seguinte seqüência: trocas na mesma rota, realocação de demanda, eliminação e combinação de rotas, inserção e adição de rotas. O procedimento de trocas na mesma rota é aplicado novamente no final do processo. O método está detalhado a seguir através de um exemplo ilustrativo. Para cada uma das fases de melhoria, consideram-se os seguintes veículos disponíveis:

Tabela 6.1 - Tipos de veículos

Tipo de veículo	Capacidade
V_1	13
V_2	17
V_3	26
V_4	49

Trocas na mesma rota

As trocas na mesma rota são efetuadas através de um movimento denominado *CYC API* (*CYClic Adjacent Pairwise Interchange*), que consiste na troca cíclica de pares adjacentes de uma determinada rota (RONCONI, 2004). Considere r uma determinada rota com p clientes, $r = 1, \dots, p$. Dado $r = (3, 2, 1, 4)$, a vizinhança $N(r)$ gerada pelo movimento *CYC API* é:

$$N(r) = \{(2, 3, 1, 4); (3, 1, 2, 4); (3, 2, 4, 1); (4, 2, 1, 3)\}$$

Este movimento é efetuado para cada rota de uma determinada solução. As trocas são efetuadas somente se houver melhoria (redução da distância) e as restrições de janelas de tempo forem respeitadas.

Realocação de demanda

Este movimento é aplicado para cada cliente i , $i = 1, \dots, n$ que está fracionado em mais de uma rota. Inicia-se com o cliente i mais distante. Para uma determinada loja i , escolhe-se, inicialmente, a rota R_j que entrega a maior quantidade Q_i , pois clientes com demandas superiores têm maior probabilidade de violar a restrição de capacidade se não forem inseridos logo no início (SALHI e RAND, 1993). Assim, calcula-se o custo de realocação da demanda da loja i da rota R_j para cada uma das rotas R_k onde a loja i está inserida e escolhe-se aquela com maior economia. A troca só é efetuada se todas as restrições do problema forem respeitadas e o custo total reduzido (soma dos custos fixos e de roteirização). O procedimento é repetido para as demais rotas. O tipo de veículo designado é sempre o menor capaz de atender a demanda da rota. Este movimento está detalhado a seguir através de um exemplo ilustrativo.

Considere um cliente $i = 4$ que está inserido em 5 rotas. Os roteiros de entrega, além da quantidade entregue a cada cliente de cada rota R_k , $k = 1, \dots, 5$, estão especificados abaixo.

$$R_1 = \left\{ \begin{matrix} 8 \\ 7, 3, 5, 4 \end{matrix} \right\}, \text{ Veículo} = V_4$$

$$R_2 = \left\{ \begin{matrix} 8 \\ 2, 4, 10 \end{matrix} \right\}, \text{ Veículo} = V_3$$

$$R_3 = \left\{ \begin{matrix} 15 \\ 4 \end{matrix} \right\}, \text{ Veículo} = V_2$$

$$R_4 = \left\{ \begin{matrix} 9 \\ 4, 3, 2 \end{matrix} \right\}, \text{ Veículo} = V_3$$

$$R_5 = \left\{ \begin{matrix} 13 \\ 1, 4, 2, 7 \end{matrix} \right\}, \text{ Veículo} = V_4$$

A maior quantidade ao cliente 4 é entregue na rota R_3 . Assim, busca-se inserir a demanda do cliente 4 da rota R_3 nas demais rotas. Na rota R_5 a capacidade do maior veículo capaz é excedida. Nas demais rotas, a maior economia ocorre na rota R_1 , já que o veículo não se altera. Como o custo total de distribuição é reduzido e as restrições do problema são respeitadas, a realocação é efetuada. Os roteiros e as quantidades entregues nas rotas passam a ser:

$$R_1 = \left\{ \begin{matrix} 8 \\ 7, 3, 5, 4 \end{matrix} \right\}, \text{ Veículo} = V_4$$

$$R_2 = \left\{ \begin{matrix} 8 \\ 2, 4, 10 \end{matrix} \right\}, \text{ Veículo} = V_3$$

$$* R_3 = \{\emptyset\}$$

$$R_4 = \left\{ \begin{matrix} 9 \\ 4, 3, 2 \end{matrix} \right\}, \text{ Veículo} = V_3$$

$$R_5 = \left\{ \begin{matrix} 13 \\ 1, 4, 2, 7 \end{matrix} \right\}, \text{ Veículo} = V_4$$

Um asterisco indica que a rota já foi testada com as demais. A próxima rota escolhida é a rota R_1 que possui a maior quantidade entregue. Os movimentos de realocação são testados nas demais rotas (R_2 , R_3 , R_4 e R_5). No caso de rotas em que já houve melhoria e o cliente i foi excluído da rota, como no caso da rota R_3 , a inserção é testada na mesma posição inicial (posição antes de iniciar o movimento realocação de demanda). A inserção do cliente 4 da rota R_1 para a rota R_3 não gerou economia. Para a rota R_5 , a restrição de capacidade do veículo é violada. Adicionando a demanda da rota R_1 nas rotas R_2 e R_4 , para ambas as rotas o veículo é alterado de V_3 para V_4 , o que faz com que as economias sejam idênticas. No caso de empate, escolhe-se a rota R_k que tiver maior demanda do cliente i . A rota R_4 , que possui maior demanda do cliente 4 ($d_4 = 9$), é então escolhida. Temos, portanto:

$$* R_1 = \left\{ \begin{matrix} 8 \\ 7, 3, 5 \end{matrix} \right\}, \text{ Veículo} = V_3$$

$$R_2 = \left\{ \begin{smallmatrix} 8 \\ 2 \end{smallmatrix}, \begin{smallmatrix} 7 \\ 4 \end{smallmatrix}, \begin{smallmatrix} 9 \\ 10 \end{smallmatrix} \right\}, \text{Veículo} = V_3$$

$$* R_3 = \{\emptyset\}$$

$$R_4 = \left\{ \begin{smallmatrix} 34 \\ 4 \end{smallmatrix}, \begin{smallmatrix} 8 \\ 3 \end{smallmatrix}, \begin{smallmatrix} 7 \\ 2 \end{smallmatrix} \right\}, \text{Veículo} = V_4$$

$$R_5 = \left\{ \begin{smallmatrix} 13 \\ 1 \end{smallmatrix}, \begin{smallmatrix} 14 \\ 4 \end{smallmatrix}, \begin{smallmatrix} 10 \\ 2 \end{smallmatrix}, \begin{smallmatrix} 5 \\ 7 \end{smallmatrix} \right\}, \text{Veículo} = V_4$$

A próxima rota escolhida é a rota R_4 (maior demanda) que não pode ser inserida nas rotas R_1 , R_2 e R_5 , pois viola a restrição de capacidade. A inserção na rota R_3 não gera economia. Portanto, as rotas não são alteradas:

$$* R_1 = \left\{ \begin{smallmatrix} 8 \\ 7 \end{smallmatrix}, \begin{smallmatrix} 10 \\ 3 \end{smallmatrix}, \begin{smallmatrix} 5 \\ 5 \end{smallmatrix} \right\}, \text{Veículo} = V_3$$

$$R_2 = \left\{ \begin{smallmatrix} 8 \\ 2 \end{smallmatrix}, \begin{smallmatrix} 7 \\ 4 \end{smallmatrix}, \begin{smallmatrix} 9 \\ 10 \end{smallmatrix} \right\}, \text{Veículo} = V_3$$

$$* R_3 = \{\emptyset\}$$

$$* R_4 = \left\{ \begin{smallmatrix} 34 \\ 4 \end{smallmatrix}, \begin{smallmatrix} 8 \\ 3 \end{smallmatrix}, \begin{smallmatrix} 7 \\ 2 \end{smallmatrix} \right\}, \text{Veículo} = V_4$$

$$R_5 = \left\{ \begin{smallmatrix} 13 \\ 1 \end{smallmatrix}, \begin{smallmatrix} 14 \\ 4 \end{smallmatrix}, \begin{smallmatrix} 10 \\ 2 \end{smallmatrix}, \begin{smallmatrix} 5 \\ 7 \end{smallmatrix} \right\}, \text{Veículo} = V_4$$

Das rotas restantes (R_2 e R_5), escolhe-se a R_5 por apresentar maior demanda. A realocação de demanda da rota R_5 para as rotas R_1 , R_2 e R_3 resulta em um aumento no custo, enquanto a rota R_4 está utilizando sua capacidade máxima. Portanto, as rotas e a quantidade entregue a cada cliente não foram alteradas:

$$* R_1 = \left\{ \begin{smallmatrix} 8 \\ 7 \end{smallmatrix}, \begin{smallmatrix} 10 \\ 3 \end{smallmatrix}, \begin{smallmatrix} 5 \\ 5 \end{smallmatrix} \right\}, \text{Veículo} = V_3$$

$$R_2 = \left\{ \begin{smallmatrix} 8 \\ 2 \end{smallmatrix}, \begin{smallmatrix} 7 \\ 4 \end{smallmatrix}, \begin{smallmatrix} 9 \\ 10 \end{smallmatrix} \right\}, \text{Veículo} = V_3$$

$$* R_3 = \{\emptyset\}$$

$$* R_4 = \left\{ \begin{smallmatrix} 34 & 8 \\ 4, & 3, & 2 \end{smallmatrix} \right\}, \text{Veículo} = V_4$$

$$* R_5 = \left\{ \begin{smallmatrix} 13 & 14 & 10 & 5 \\ 1, & 4, & 2, & 7 \end{smallmatrix} \right\}, \text{Veículo} = V_4$$

Restou apenas a rota R_2 que pode ser inserida nas rotas R_1 , R_3 e R_5 , pois a rota R_4 já está utilizando sua capacidade máxima. A realocação da demanda da rota R_2 para a rota R_5 resulta em melhoria. O resultado final é, portanto:

$$* R_1 = \left\{ \begin{smallmatrix} 8 & 10 & 5 \\ 7, & 3, & 5 \end{smallmatrix} \right\}, \text{Veículo} = V_3$$

$$* R_2 = \left\{ \begin{smallmatrix} 8 & 9 \\ 2, & 10 \end{smallmatrix} \right\}, \text{Veículo} = V_2$$

$$* R_3 = \{\emptyset\}$$

$$* R_4 = \left\{ \begin{smallmatrix} 34 & 8 & 7 \\ 4, & 3, & 2 \end{smallmatrix} \right\}, \text{Veículo} = V_4$$

$$* R_5 = \left\{ \begin{smallmatrix} 13 & 21 & 10 & 5 \\ 1, & 4, & 2, & 7 \end{smallmatrix} \right\}, \text{Veículo} = V_4$$

Eliminação e combinação de rotas

Esta fase busca eliminar rotas com veículos menores que o segundo maior veículo da frota. O critério de eliminação também pode ser aplicado para rotas com n ou menos clientes ou rotas com capacidade ociosa. Para cada possível rota R_j eliminada (inicia-se pela rota mais longa, ou seja, a que possui a maior distância total percorrida), busca-se combiná-la com outra rota (R_k), escolhida através de uma lista candidata. A lista candidata é baseada nas idéias de Corberán et al. (2002) e está descrita a seguir.

Para cada rota avaliada (R_j), constrói-se uma lista candidata, com base nos melhores pares de rotas ((R_j, R_k)). O melhor par é aquele que minimiza a distância entre duas rotas. Quando as rotas de um dado par apresentam apenas um cliente, a distância entre elas é simplesmente a distância entre os dois clientes. Quando uma rota tem mais de um cliente,

consideram-se os pontos extremos para calcular a distância mínima. A distância mínima entre as rotas R_j e R_k é dada por:

$$\min \left\{ d_{r_j(1)r_k(n_k)}, d_{r_k(1)r_j(n_j)} \right\}$$

onde:

$d_{r_j(1)r_k(n_k)}$ é a distância do primeiro cliente da rota R_j com o último cliente da rota R_k

$d_{r_k(1)r_j(n_j)}$ é a distância do primeiro cliente da rota R_k com o último cliente da rota R_j .

No movimento de combinação, todos os clientes da rota R_j serão testados em todas as posições da rota R_k . Inicia-se com o primeiro cliente da rota R_j . A combinação de rotas é efetuada somente se houver melhoria (redução dos custos fixos e de roteirização) e todas as restrições do problema forem respeitadas, após inserção de todos os clientes da rota R_j . O algoritmo pode ser expresso da seguinte forma:

Figura 6.6 - Algoritmo de eliminação e combinação de rotas

Para cada possível rota R_j a ser eliminada (inicia-se pela rota mais longa)

Escolher uma rota R_k entre as quatro primeiras rotas da lista candidata. Verificar se a rota combinada (junção dos clientes das rotas R_j e R_k) é viável em termos de capacidade e tipo de veículo. Se a resposta for positiva, passar para o próximo passo. Caso contrário, repetir o procedimento para as próximas quatro rotas da lista.

Inserir cada cliente da rota R_j (iniciar pelo primeiro cliente) em todas as posições da rota R_k e escolher a melhor posição viável (que minimiza o acréscimo de custo e tempo e atenda a restrição de janela de tempo).

Se, após inserção de todos os clientes da rota R_j na rota R_k , houver melhoria (custo fixo e custos de roteirização) e as restrições forem respeitadas, a combinação é efetuada.

Inserção

O movimento de inserção é implementado para cada rota R_j , onde todos os clientes viáveis serão testados em todas as posições de uma outra rota R_k , escolhida através de uma

lista candidata, semelhante à lista descrita no movimento de eliminação e combinação de rotas. Inicia-se com a rota R_j mais longa, que é aquela que possui a maior distância total percorrida e, para cada rota R_j escolhida, inicia-se com o cliente i mais distante. A inserção de um determinado cliente é efetuada somente se houver melhoria (custo fixo e custo de roteirização) e todas as restrições do problema forem respeitadas.

O algoritmo pode ser expresso da seguinte forma:

Figura 6.7 - Algoritmo de inserção

Para cada rota R_j (inicia-se com a rota mais longa)

Escolher uma rota, aleatoriamente, entre as quatro primeiras rotas R_k da lista candidata. Verificar se pelo menos uma loja da rota R_j é viável em termos de capacidade e tipo de veículo (após inserção na rota R_k). Se a resposta for positiva, passar para o passo seguinte. Caso contrário, aplicar o mesmo procedimento para as próximas quatro rotas da lista.

Inserir cada cliente viável da rota R_j (iniciar pelo cliente mais distante) em todas as posições da rota R_k e escolher a melhor delas, caso haja melhoria (custo fixo e custo de roteirização) e as restrições sejam respeitadas.

Adição de rotas

Da mesma forma que na realocação de demanda, consideram-se todos os clientes i cuja demanda está fracionada em mais de uma rota. Inicia-se com o cliente i mais distante. Para um determinado cliente i , inicia-se com a rota R_j que entrega a menor quantidade. Neste movimento, caso a demanda do cliente i da rota R_j seja maior que a capacidade do maior caminhão, o mesmo é abastecido enquanto houver espaço e a demanda remanescente é adicionada a um novo veículo (menor veículo capaz). Para cada rota R_j avaliada, adiciona-se a demanda do cliente i a uma nova rota ou a rota existente R_k e calcula-se o acréscimo ou redução no custo total (soma dos custos fixos e de roteirização) após esse movimento. No final, escolhe-se a melhor combinação, caso haja economia.

Consideremos o mesmo exemplo do movimento realocação de demanda, onde o cliente $i = 4$ está inserido em 5 rotas:

$$R_1 = \left\{ \begin{matrix} 8 \\ 7 \end{matrix}, \begin{matrix} 11 \\ 3 \end{matrix}, \begin{matrix} 5 \\ 5 \end{matrix}, \begin{matrix} 10 \\ 4 \end{matrix} \right\}, \text{Veículo} = V_4$$

$$R_2 = \left\{ \begin{matrix} 8 \\ 2 \end{matrix}, \begin{matrix} 7 \\ 4 \end{matrix}, \begin{matrix} 9 \\ 10 \end{matrix} \right\}, \text{Veículo} = V_3$$

$$R_3 = \left\{ \begin{matrix} 15 \\ 4 \end{matrix} \right\}, \text{Veículo} = V_2$$

$$R_4 = \left\{ \begin{matrix} 9 \\ 4 \end{matrix}, \begin{matrix} 8 \\ 3 \end{matrix}, \begin{matrix} 7 \\ 2 \end{matrix} \right\}, \text{Veículo} = V_3$$

$$R_5 = \left\{ \begin{matrix} 13 \\ 1 \end{matrix}, \begin{matrix} 14 \\ 4 \end{matrix}, \begin{matrix} 10 \\ 2 \end{matrix}, \begin{matrix} 5 \\ 7 \end{matrix} \right\}, \text{Veículo} = V_4$$

1) A primeira rota escolhida é R_2 que entrega a menor quantidade para o cliente 4.

Adicionou-se uma nova rota R_6 com o menor veículo capaz de atender esta demanda, que é excluída da rota R_2 . Este movimento gerou um aumento no custo total de R\$50,00. Temos:

$$R_1 = \left\{ \begin{matrix} 8 \\ 7 \end{matrix}, \begin{matrix} 11 \\ 3 \end{matrix}, \begin{matrix} 5 \\ 5 \end{matrix}, \begin{matrix} 10 \\ 4 \end{matrix} \right\}, \text{Veículo} = V_4$$

$$* R_2 = \left\{ \begin{matrix} 8 \\ 2 \end{matrix}, \begin{matrix} 9 \\ 10 \end{matrix} \right\}, \text{Veículo} = V_2$$

$$R_3 = \left\{ \begin{matrix} 15 \\ 4 \end{matrix} \right\}, \text{Veículo} = V_2$$

$$R_4 = \left\{ \begin{matrix} 9 \\ 4 \end{matrix}, \begin{matrix} 8 \\ 3 \end{matrix}, \begin{matrix} 7 \\ 2 \end{matrix} \right\}, \text{Veículo} = V_3$$

$$R_5 = \left\{ \begin{matrix} 13 \\ 1 \end{matrix}, \begin{matrix} 14 \\ 4 \end{matrix}, \begin{matrix} 10 \\ 2 \end{matrix}, \begin{matrix} 5 \\ 7 \end{matrix} \right\}, \text{Veículo} = V_4$$

$$R_6 = \left\{ \begin{matrix} 7 \\ 4 \end{matrix} \right\}, \text{Veículo} = V_1$$

2) A segunda rota escolhida é R_4 que entrega a menor quantidade. A quantidade entregue ao cliente 4 na rota R_4 ($Q_4^4 = 9$) foi transferida para a rota R_6 e o veículo é alterado

para V_2 . Esta troca também gerou um aumento no custo total, porém, de apenas R\$20,00.

Temos, portanto:

$$R_1 = \left\{ \begin{matrix} 8 \\ 7 \end{matrix}, \begin{matrix} 11 \\ 3 \end{matrix}, \begin{matrix} 5 \\ 5 \end{matrix}, \begin{matrix} 10 \\ 4 \end{matrix} \right\}, \text{Veículo} = V_4$$

$$* R_2 = \left\{ \begin{matrix} 8 \\ 2 \end{matrix}, \begin{matrix} 9 \\ 10 \end{matrix} \right\}, \text{Veículo} = V_2$$

$$R_3 = \left\{ \begin{matrix} 15 \\ 4 \end{matrix} \right\}, \text{Veículo} = V_2$$

$$* R_4 = \left\{ \begin{matrix} 8 \\ 3 \end{matrix}, \begin{matrix} 7 \\ 2 \end{matrix} \right\}, \text{Veículo} = V_2$$

$$R_5 = \left\{ \begin{matrix} 13 \\ 1 \end{matrix}, \begin{matrix} 14 \\ 4 \end{matrix}, \begin{matrix} 10 \\ 2 \end{matrix}, \begin{matrix} 5 \\ 7 \end{matrix} \right\}, \text{Veículo} = V_4$$

$$R_6 = \left\{ \begin{matrix} 16 \\ 4 \end{matrix} \right\}, \text{Veículo} = V_2$$

3) A terceira rota escolhida é R_1 que entrega a menor quantidade. A quantidade entregue ao cliente 4 na rota R_1 ($Q_4^1 = 10$) foi transferida para a rota R_6 e o veículo é alterado para V_3 . Esta troca gerou uma redução no custo total de R\$50,00:

$$* R_1 = \left\{ \begin{matrix} 8 \\ 7 \end{matrix}, \begin{matrix} 11 \\ 3 \end{matrix}, \begin{matrix} 5 \\ 5 \end{matrix} \right\}, \text{Veículo} = V_3$$

$$* R_2 = \left\{ \begin{matrix} 8 \\ 2 \end{matrix}, \begin{matrix} 9 \\ 10 \end{matrix} \right\}, \text{Veículo} = V_2$$

$$R_3 = \left\{ \begin{matrix} 15 \\ 4 \end{matrix} \right\}, \text{Veículo} = V_2$$

$$* R_4 = \left\{ \begin{matrix} 8 \\ 3 \end{matrix}, \begin{matrix} 7 \\ 2 \end{matrix} \right\}, \text{Veículo} = V_2$$

$$R_5 = \left\{ \begin{matrix} 13 \\ 1 \end{matrix}, \begin{matrix} 14 \\ 4 \end{matrix}, \begin{matrix} 10 \\ 2 \end{matrix}, \begin{matrix} 5 \\ 7 \end{matrix} \right\}, \text{Veículo} = V_4$$

$$R_6 = \left\{ \begin{matrix} 26 \\ 4 \end{matrix} \right\}, \text{Veículo} = V_3$$

4) A quarta rota escolhida é R_5 . A demanda entregue ao cliente 4 na rota R_5 ($Q_4^5 = 14$) foi transferida para a rota R_6 . O veículo foi alterado para V_4 , gerando uma economia de R\$20,00:

$$* R_1 = \left\{ \begin{matrix} 8 \\ 7, 3, 5 \end{matrix} \right\}, \text{Veículo} = V_3$$

$$* R_2 = \left\{ \begin{matrix} 8 \\ 2, 10 \end{matrix} \right\}, \text{Veículo} = V_2$$

$$R_3 = \left\{ \begin{matrix} 15 \\ 4 \end{matrix} \right\}, \text{Veículo} = V_2$$

$$* R_4 = \left\{ \begin{matrix} 8 \\ 3, 2 \end{matrix} \right\}, \text{Veículo} = V_2$$

$$* R_5 = \left\{ \begin{matrix} 13 \\ 1, 2, 7 \end{matrix} \right\}, \text{Veículo} = V_4$$

$$R_6 = \left\{ \begin{matrix} 40 \\ 4 \end{matrix} \right\}, \text{Veículo} = V_4$$

5) Finalmente, para atender a demanda da rota R_3 , é necessária a adição de uma nova rota, pois a capacidade do veículo atual é excedida. A demanda é adicionada, primeiramente, na rota R_6 até atingir a capacidade máxima do veículo e a demanda remanescente é adicionada a uma nova rota (menor veículo capaz). Neste caso (apenas no maior veículo capaz), verificam-se 2 possibilidades: a) o acréscimo ou redução de custo com a adição da rota atual (R_6) até atingir sua capacidade máxima; b) acréscimo ou redução de custo com a adição de uma nova rota (R_7) para atender a demanda remanescente.

a) O primeiro movimento gerou uma economia de R\$40,00 no custo total. Temos:

$$* R_1 = \left\{ \begin{matrix} 8 \\ 7, 3, 5 \end{matrix} \right\}, \text{Veículo} = V_3$$

$$* R_2 = \left\{ \begin{matrix} 8 \\ 2, 10 \end{matrix} \right\}, \text{Veículo} = V_2$$

$$* R_3 = \left\{ \begin{matrix} 6 \\ 4 \end{matrix} \right\}, \text{Veículo} = V_1$$

$$* R_4 = \left\{ \begin{smallmatrix} 8 \\ 3, 2 \end{smallmatrix} \right\}, \text{ Veículo} = V_2$$

$$* R_5 = \left\{ \begin{smallmatrix} 13 & 10 \\ 1, 2, 7 \end{smallmatrix} \right\}, \text{ Veículo} = V_4$$

$$R_6' = \left\{ \begin{smallmatrix} 49 \\ 4 \end{smallmatrix} \right\}, \text{ Veículo} = V_4$$

b) Os casos a e b, coincidentemente, geraram o mesmo resultado. Portanto, o segundo movimento também gerou uma economia de R\$40,00 no custo total. Temos:

$$* R_1 = \left\{ \begin{smallmatrix} 8 & 11 & 5 \\ 7, 3, 5 \end{smallmatrix} \right\}, \text{ Veículo} = V_3$$

$$* R_2 = \left\{ \begin{smallmatrix} 8 & 9 \\ 2, 10 \end{smallmatrix} \right\}, \text{ Veículo} = V_2$$

$$* R_3 = \{\emptyset\}$$

$$* R_4 = \left\{ \begin{smallmatrix} 8 & 7 \\ 3, 2 \end{smallmatrix} \right\}, \text{ Veículo} = V_2$$

$$* R_5 = \left\{ \begin{smallmatrix} 13 & 10 & 5 \\ 1, 2, 7 \end{smallmatrix} \right\}, \text{ Veículo} = V_4$$

$$R_6' = \left\{ \begin{smallmatrix} 49 \\ 4 \end{smallmatrix} \right\}, \text{ Veículo} = V_4$$

$$R_7' = \left\{ \begin{smallmatrix} 6 \\ 4 \end{smallmatrix} \right\}, \text{ Veículo} = V_1$$

Verificadas todas as possibilidades, escolhe-se aquela com maior economia, que é o caso 3. Portanto, a demanda da loja 4 é excluída das rotas R_1 , R_2 e R_4 e adicionada a uma nova rota R_6 :

$$R_6 = \left\{ \begin{smallmatrix} 26 \\ 4 \end{smallmatrix} \right\}, \text{ Tipo do veículo} = V_3; \Delta = - R\$50,00.$$

Trocas na mesma rota

O procedimento de trocas na mesma rota é repetido ao final das melhorias.

3. Método de criação e atualização do conjunto de referência

3.1 Criação do conjunto de referência

A partir do conjunto de P soluções diversas, extrai-se o conjunto de referência (b). O conjunto de referência inicia-se com as b_1 soluções de alta qualidade e as b_2 soluções restantes são extraídas de P com o objetivo de maximizar a distância mínima em relação às soluções incluídas no conjunto de referência. O critério de distância utilizado está descrito ao final do método de combinação.

3.2 Atualização do conjunto de referência

São implementadas atualização estática, onde o conjunto de referência não muda até que todas as combinações de soluções de *RefSet* tenham sido realizadas, e atualização dinâmica, no qual o conjunto de referência é atualizado assim que uma solução combinada gere melhoria. Os critérios de atualização utilizados são por qualidade (Q) e qualidade e diversidade (QD).

4. Método de geração de subconjuntos

O método de combinação é aplicado para todos os pares de soluções do conjunto de referência (combinações do tipo 1).

5. Método de combinação das soluções

Este método consiste na geração de novas soluções através da combinação de duas soluções de *RefSet*. O método de combinação foi baseado nas idéias de Corberán et al. (2002) e Rego e Leão (2000). O algoritmo implementado está detalhado a seguir, através de um exemplo ilustrativo.

Tomemos um exemplo hipotético que trata de um problema de roteirização de veículos com entregas fracionadas, com $n = 10$ clientes e 3 tipos de veículos. A capacidade dos veículos e a demanda dos clientes estão apresentadas nas tabelas abaixo.

Tabela 6.2 - Capacidade dos veículos

Veículo	Capacidade
V₁	49
VB₂	26
V₃	13

Tabela 6.3 - Demanda dos clientes

Cliente	Demanda
1	20
2	30
3	22
4	15
5	16
6	8
7	13
8	55
9	19
10	10

Duas soluções (A e B) foram obtidas para este problema. Assume-se que A_k é a k -ésima rota da solução A ($k = 1, \dots, m$) e B_k é a k -ésima rota da solução B ($k = 1, \dots, n$). O tipo de veículo é o menor capaz de atender a demanda total da rota. Em cada solução, além dos roteiros, está determinada a quantidade entregue a cada cliente da rota:

$$A - A_1 = \begin{Bmatrix} 8 & 8 & 10 \\ 8, 4, 1 \end{Bmatrix}, \quad A_2 = \begin{Bmatrix} 9 & 7 & 10 \\ 8, 4, 1 \end{Bmatrix}, \quad A_3 = \begin{Bmatrix} 26 \\ 8 \end{Bmatrix}, \quad A_4 = \begin{Bmatrix} 12 \\ 8 \end{Bmatrix}, \quad A_5 = \begin{Bmatrix} 15 & 7 \\ 2, 7 \end{Bmatrix}, \quad A_6 = \begin{Bmatrix} 15 & 6 \\ 2, 7 \end{Bmatrix}, \quad A_7 = \begin{Bmatrix} 22 & 16 & 8 \\ 3, 5, 6 \end{Bmatrix},$$

$$A_8 = \begin{Bmatrix} 19 \\ 9 \end{Bmatrix} \text{ e } A_9 = \begin{Bmatrix} 10 \\ 10 \end{Bmatrix}.$$

$$B - B_1 = \begin{Bmatrix} 14 & 15 & 20 \\ 8, 4, 1 \end{Bmatrix}, \quad B_2 = \begin{Bmatrix} 41 \\ 8 \end{Bmatrix}, \quad B_3 = \begin{Bmatrix} 30 & 13 \\ 2, 7 \end{Bmatrix}, \quad B_4 = \begin{Bmatrix} 22 & 16 & 6 \\ 3, 5, 9 \end{Bmatrix}, \quad B_5 = \begin{Bmatrix} 13 \\ 9 \end{Bmatrix}, \quad B_6 = \begin{Bmatrix} 8 \\ 6 \end{Bmatrix} \text{ e } B_7 = \begin{Bmatrix} 10 \\ 10 \end{Bmatrix}.$$

O método de combinação é dividido em duas fases. A fase 1 tem como objetivo combinar apenas os elementos comuns das rotas combinadas e a fase 2 atender a demanda remanescente de todos os clientes.

FASE 1

O procedimento de combinação da fase 1 é construído a partir de uma matriz $A \times B$, onde os componentes (A_k, B_k) da matriz contêm o número de elementos comuns entre as rotas $k = 1, \dots, m$ da solução A e $k = 1, \dots, n$ da solução B . A tabela 6.4 apresenta a matriz de combinação.

Tabela 6.4 - Matriz de combinação

	B_1	B_2	B_3	B_4	B_5	B_6	B_7
A_1	3	1	0	0	0	0	0
A_2	3	1	0	0	0	0	0
A_3	1	1	0	0	0	0	0
A_4	1	1	0	0	0	0	0
A_5	0	0	2	0	0	0	0
A_6	0	0	2	0	0	0	0
A_7	0	0	0	2	0	1	0
A_8	0	0	0	0	1	0	0
A_9	0	0	0	0	0	0	1

Primeiramente definem-se quais rotas de cada solução são combinadas. Esta fase inicia com a combinação dos componentes que possuem o maior número de elementos

comuns. De acordo com a tabela 6.4, podemos concluir que as combinações $A_1 \cap B_1$ e $A_2 \cap B_1$ possuem o maior número de elementos em comum. No caso de empate escolhe-se a combinação de rotas que minimiza a função f , tal que as diferenças entre as quantidades entregues aos clientes em comum seja mínima:

$$\min f = \sum_{i \in k} |Q_{i,k}^A - Q_{i,k}^B| :$$

onde i é o elemento em comum da rota $k = 1, \dots, m$ da solução A e da rota $k = 1, \dots, n$ da solução B ; $Q_{i,k}^A$ é a quantidade entregue ao cliente i da rota $k = 1, \dots, m$ da solução A e $Q_{i,k}^B$ é a quantidade entregue ao cliente i da rota $k = 1, \dots, n$ da solução B .

Neste exemplo, o valor de f é semelhante para $A_1 \cap B_1$ ($|8-14|+|8-15|+|10-20|=23$) e para $A_2 \cap B_1$ ($|9-14|+|7-15|+|10-20|=23$). Assim, escolhe-se uma das combinações aleatoriamente que, neste caso, foi $A_1 \cap B_1$.

A rota combinada é formada pelos elementos comuns das rotas (8, 4, 1). O próximo passo consiste em determinar a quantidade entregue aos elementos da rota combinada e o tipo de veículo. A quantidade entregue para cada elemento é a menor quantidade entre as duas soluções combinadas e o tipo de veículo é o menor veículo capaz de atender a demanda da rota. O roteiro da solução combinada é semelhante ao da melhor solução e, no caso de empate, escolhe-se o roteiro de uma das soluções aleatoriamente. Desta forma temos:

$$R_1 = \left\{ \begin{matrix} 8 & 8 & 10 \\ 8, 4, 1 \end{matrix} \right\}, \text{ e o veículo designado é } V_2.$$

As rotas A_1 e B_1 não podem mais ser combinadas. A próxima combinação de rotas com maior número de elementos em comum é $A_5 \cap B_3$, $A_6 \cap B_3$ e $A_7 \cap B_4$. O valor de f para a combinação A_5 e B_3 é 21 ($|15-30|+|7-13|$). Já entre as rotas A_6 e B_3 é 22 ($|15-30|+|6-13|$) e entre as rotas A_7 e B_4 é $|22-22|+|16-16|=0$ (consideram-se apenas os

elementos em comum, que neste caso é 3 e 5). Portanto, a segunda combinação é referente às rotas A_7 e B_4 :

$$R_2 = \left\{ \begin{smallmatrix} 22 & 16 \\ 3, 5 \end{smallmatrix} \right\}, \text{ e o veículo designado é } V_1.$$

As rotas A_7 e B_4 não podem mais ser combinadas. A próxima combinação é das rotas A_5 e B_3 , que possuem o maior número de elementos comuns e o menor valor de $f(21)$. A rota combinada é:

$$R_3 = \left\{ \begin{smallmatrix} 15 & 7 \\ 2, 7 \end{smallmatrix} \right\}, \text{ e o veículo designado é } V_2.$$

As rotas A_5 e B_3 não podem mais ser combinadas. As próximas combinações de rotas com maior número de elementos comuns são $A_2 \cap B_2$, $A_3 \cap B_2$, $A_4 \cap B_2$, $A_8 \cap B_5$ e $A_9 \cap B_7$, todas com 1 elemento em comum. As rotas A_9 e B_7 possuem o mesmo elemento em comum e a mesma quantidade entregue, ou seja, $f = 0$. Portanto, a próxima combinação $A_9 \cap B_7$ é:

$$R_4 = \left\{ \begin{smallmatrix} 10 \\ 10 \end{smallmatrix} \right\}, \text{ e o veículo designado é } V_3.$$

A próxima combinação é $A_8 \cap B_5$, no qual o valor de f é 6. A rota combinada é:

$$R_5 = \left\{ \begin{smallmatrix} 13 \\ 9 \end{smallmatrix} \right\}, \text{ e o veículo é } V_3.$$

O valor de f para a combinação $A_2 \cap B_2$ é 32 ($|9 - 41|$), para $A_3 \cap B_2$ é 15 ($|26 - 41|$) e para $A_4 \cap B_2$ é 29 ($|12 - 41|$). Portanto, a combinação escolhida é $A_3 \cap B_2$:

$$R_6 = \left\{ \begin{smallmatrix} 26 \\ 8 \end{smallmatrix} \right\}, \text{ e o veículo é } V_2.$$

FASE 2

Esta fase tem como objetivo atender a demanda remanescente dos clientes que já foram atendidos por alguma rota na fase 1, e atender a demanda total dos clientes que ainda não pertencem a nenhuma rota. Isto é feito através de um procedimento de inserção. O cliente i a ser inserido é o cliente mais distante. Primeiramente, verifica-se se ele já pertence a alguma rota combinada. A partir daí, dois passos podem ser tomados:

PASSO A

Caso o cliente i já pertença a pelo menos uma rota combinada (fase 1 do método de combinação), escolhe-se aquela com maior capacidade ociosa e entrega-se o mínimo entre a capacidade ociosa do veículo atual e a demanda do cliente i . Enquanto a demanda total do cliente i não for atendida e houver uma rota com capacidade ociosa em que o cliente i está inserido, o procedimento é repetido (sem mudar o veículo atual).

Caso o cliente i pertença a apenas uma rota combinada e a demanda total não tenha sido atendida com o veículo atual, o veículo é alterado para o menor capaz de atender a demanda remanescente. Caso a demanda remanescente do cliente i seja maior que a capacidade ociosa do maior veículo capaz de atendê-lo, a demanda é adicionada enquanto houver capacidade e a demanda remanescente será entregue através do passo B.

Se o cliente i pertencer a mais de uma rota combinada e a demanda total não for atendida com os veículos atuais, entrega-se o mínimo entre a capacidade ociosa do maior veículo capaz e a demanda remanescente do cliente i , e a rota escolhida é aquela que minimiza o acréscimo de custo fixo. O procedimento é repetido até que a demanda total do cliente i tenha sido atendida ou até que todos os veículos (maiores capazes) no qual o cliente i está inserido estejam utilizando sua capacidade máxima.

Ao final deste passo, caso a demanda total do cliente i ainda não tenha sido atendida, aplica-se o passo B. Caso contrário, escolhe-se o próximo cliente mais distante a ser inserido.

PASSO B

Caso o cliente i não pertença a nenhuma rota combinada ou a demanda total do cliente i não tenha sido atendida através do passo A, aplica-se o passo B.

Com base nas soluções iniciais A e B , verificam-se todas as rotas em que o cliente i está inserido e todos os arcos no qual $x_{ij} = 1$ ou $x_{ji} = 1$. Assim, para cada j pertencente a uma das rotas combinadas no passo A, com exceção do depósito ($j = 0$), calcula-se o custo de inserção do cliente i (acréscimo de custo fixo do veículo, custo de roteirização e tempo) nas posições x_{ij} (quando $x_{ij} = 1$) ou x_{ji} (quando $x_{ji} = 1$). Considera-se $x_{i0} = 1$ ou $x_{0i} = 1$ apenas para rotas com carga completa ($0 - i - 0$).

Para cada possível posição de inserção do cliente i , entrega-se o mínimo entre a capacidade ociosa do maior veículo capaz e a demanda do cliente i , e escolhe-se aquela com menor custo, desde que as restrições de janelas de tempo e tipo de veículo sejam respeitadas. O procedimento é repetido até que a demanda total do cliente i seja atendida ou enquanto houver uma posição de inserção do cliente i .

Verificadas todas as rotas das soluções A e B em que o cliente i está inserido, caso a demanda total não tenha sido atendida, adiciona-se uma nova rota (menor rota capaz de atender a demanda remanescente do cliente i). O procedimento é repetido até que a demanda total do cliente i seja atendida.

O método de combinação descrito busca fracionar a demanda dos clientes em diversas rotas, com base nas estratégias utilizadas na fase 1 e nos passos A e B da fase 2. Porém, caso esta não seja a melhor estratégia, o método de melhoria se encarrega de melhorar a solução combinada, através dos movimentos de realocação de demanda e adição de rotas. Em função disto, melhorias baseadas no fracionamento da demanda não foram implementadas.

De acordo com o exemplo acima, a tabela 6.5 mostra a lista em ordem decrescente dos clientes mais distantes e suas respectivas demandas remanescentes.

Tabela 6.5 – Demanda remanescente

Cliente mais distante	Demanda remanescente
1	10
4	7
8	21
7	6
2	15
6	8
9	6

O primeiro cliente a ser inserido é o cliente 1, que pertence apenas à rota R_1 . O veículo atual é o V_2 , que está utilizando sua capacidade máxima. Como todos os clientes da rota podem receber o maior veículo V_1 , o mesmo é alterado e a quantidade entregue ao cliente 1 nesta rota passa a ser 20. O próximo cliente inserido é o cliente 4, que também pertence à rota R_1 . A quantidade entregue ao cliente 4 é alterada para 15.

O terceiro cliente da lista é o cliente 8, que pertence à rota R_1 e R_6 . A capacidade do veículo V_1 da rota R_1 é 49 e ele está utilizando uma capacidade de 43, havendo, portanto, uma capacidade ociosa. Já na rota R_6 o veículo está utilizando sua capacidade máxima. A primeira rota escolhida para inserção da demanda remanescente do cliente 8 é, portanto, a rota R_1 . Porém, a quantidade máxima que pode ser entregue é 6. Como não há nenhuma outra rota em que o cliente 8 está inserido, a demanda remanescente do cliente 8 ($d_8^{\text{rem}} = 15$) é entregue na rota R_6 e o veículo é alterado para V_1 . Temos, portanto:

$$R_1 = \left\{ \begin{matrix} 14 & 15 & 20 \\ 8, 4, 1 \end{matrix} \right\}, \text{ com veículo } V_1 \quad \text{e} \quad R_6 = \left\{ \begin{matrix} 41 \\ 8 \end{matrix} \right\}, \text{ com veículo } V_1.$$

O próximo cliente da lista é o cliente 7, que já pertence à rota R_3 . O veículo atual desta rota é V_2 e, para que a demanda remanescente do cliente 7 seja atendida nesta rota, o veículo precisaria ser alterado para V_1 . Como os clientes 2 e 7 podem receber o novo veículo, a demanda remanescente do cliente 7 é entregue nesta rota. Da mesma forma, a demanda remanescente do cliente 2 (próximo cliente mais distante) também é entregue na rota R_3 :

$$R_3 = \left\{ \begin{smallmatrix} 30 & 13 \\ 2, 7 \end{smallmatrix} \right\}, \text{ com veículo } V_1.$$

O próximo cliente da lista é 6, que não pertence à nenhuma rota combinada (passo B). Na solução A o cliente 6 está ligado ao cliente 5 na rota R_7 ($x_{56} = 1$) e, na solução B, apenas ao depósito (x_{06} ou $x_{60} = 1$). O custo de inserção do cliente 6 na última posição da rota R_2 (rota combinada em que está inserido o cliente 5, tal que $x_{56} = 1$) é menor do que a adição de uma nova rota (x_{06} ou $x_{60} = 1$). Desta forma, a rota R_2 passa a ser:

$$R_2 = \left\{ \begin{smallmatrix} 22 & 16 & 8 \\ 3, 5, 6 \end{smallmatrix} \right\}, \text{ com veículo } V_1.$$

Finalmente, para o cliente 9 que já está inserido apenas na rota R_5 , a quantidade entregue na rota passa a ser 19 e o veículo é alterado de V_1 para V_2 :

$$R_5 = \left\{ \begin{smallmatrix} 19 \\ 9 \end{smallmatrix} \right\}, \text{ com veículo } V_2.$$

A solução combinada é:

$$R_1 = \left\{ \begin{smallmatrix} 14 & 15 & 20 \\ 8, 4, 1 \end{smallmatrix} \right\}, R_2 = \left\{ \begin{smallmatrix} 22 & 16 & 8 \\ 3, 5, 6 \end{smallmatrix} \right\}, R_3 = \left\{ \begin{smallmatrix} 30 & 13 \\ 2, 7 \end{smallmatrix} \right\}, R_4 = \left\{ \begin{smallmatrix} 10 \\ 10 \end{smallmatrix} \right\}, R_5 = \left\{ \begin{smallmatrix} 19 \\ 9 \end{smallmatrix} \right\}, R_6 = \left\{ \begin{smallmatrix} 41 \\ 8 \end{smallmatrix} \right\}.$$

O método de combinação garante que as soluções finais sejam factíveis.

O critério de distância é calculado pela soma do número de arcos não-comuns de cada solução antes da combinação. Caso um arco pertença a mais de uma rota, soma-se uma unidade para cada arco não-comum. Considera-se o arco x_{i_0} ou x_{0_i} apenas para rotas com carga completa ($0 - i - 0$).

A solução A apresenta os seguintes arcos: 8–4; 4–1; 8–4; 4–1; 0–8; 0–8; 2–7; 2–7; 3–5; 5–6; 0–9; 0–10. Já a solução B apresenta os seguintes arcos: 8–4; 4–1; 0–8; 2–7; 3–5; 5–9; 0–9; 0–6; 0–10. Os arcos não-comuns da solução A são: 8–4; 4–1; 0–8; 2–7; 5–6 e da solução B são 5–9; 0–6. Portanto, a distância das soluções A e B é 7.

Para cada solução combinada de *RefSet* aplica-se o método de melhoria descrito no item 2. No final deste passo, o conjunto de referência é atualizado e, caso haja pelo menos uma nova solução no *RefSet*, o método de combinação é aplicado novamente. Caso contrário, completa-se uma iteração e o conjunto de referência é reconstruído, selecionando as b_1 melhores soluções de *RefSet* e excluindo as b_2 piores soluções. Um novo conjunto de P soluções diversas é gerado através do método de diversificação e o conjunto de soluções diversas b_2 é reconstruído. O algoritmo pára ao final de 5 iterações ou ao atingir o tempo máximo de 1 hora, até que a última iteração seja finalizada. Cabe lembrar que, para as 3 primeiras iterações, utiliza-se o método de diversificação GRACO1 e, a partir da quarta iteração, o método GRACO2.

6.6 INSTÂNCIAS DE PROBLEMAS DA LITERATURA

O problema estudado neste trabalho (HFVRPTWSD) é uma combinação do problema de roteirização de veículos com frota heterogênea (HFVRP), do problema de roteirização de veículos com janelas de tempo (VRPTW) e do problema de roteirização com demanda fracionada (VRPSD). Como o número de veículos é ilimitado, o algoritmo também é capaz de resolver o problema de dimensionamento e composição de uma frota heterogênea de veículos. Assim, com o objetivo de verificar o desempenho dos algoritmos em relação às características do HFVRPTWSD, os algoritmos foram aplicados em diversas instâncias de problemas encontradas na literatura:

- 1) Instâncias de problemas de Solomon (1987), para problemas de roteirização de veículos com janelas de tempo (VRPTW)

- 2) Instâncias de problemas de Ho e Haugland (2004), para problemas de roteirização de veículos com janelas de tempo e entregas fracionadas (VRPTWSD)
- 3) Instâncias de problemas de Liu e Shen (1999), para problemas de dimensionamento e roteirização de frota heterogênea com janelas de tempo (FSMVRPTW)

Como as três instâncias não apresentam todas as características do problema estudado neste trabalho, e sim, parte delas, são feitas algumas adaptações em relação às heurísticas construtivas e à metaheurística *scatter search* para cada conjunto de problemas. Com o objetivo de facilitar o entendimento do algoritmo, alguns passos são repetidos, tanto nas instâncias de Solomon (1987), quanto nas instâncias de Ho e Haugland (2004) e Liu e Shen (1999).

6.6.1 INSTÂNCIAS DE PROBLEMAS DE SOLOMON (1987)

Conforme dito anteriormente, o conjunto de problemas de Solomon é composto por 100 clientes. A porcentagem de clientes com janela de tempo varia entre 25, 50, 75 e 100% conforme o problema. O autor considera seis conjuntos de problemas: R1, R2, C1, C2, RC1 e RC2. Nos conjuntos R1 e R2 a posição dos clientes é gerada aleatoriamente através de uma distribuição uniforme. Nos conjuntos C1 e C2 os clientes estão localizados em grupos. Nos conjuntos RC1 e RC2, os clientes estão em semi-agrupamentos, ou seja, parte dos clientes está localizada aleatoriamente e parte em grupos. Além disso, os problemas R1, C1 e RC1 possuem um horizonte de planejamento curto e, combinados com veículos de menor capacidade, permitem somente que alguns clientes (3-8) possam ser servidos pelo mesmo veículo. Os conjuntos R2, C2 e RC2 têm um horizonte de planejamento longo e, por possuírem veículos de maior capacidade, são capazes de atender mais do que 10 clientes por rota.

Em cada conjunto de problemas, as distribuições geográficas dos clientes, a demanda e o tempo de serviço não se alteram. Desta forma, no conjunto R1, os problemas R101 a R104 são idênticos, com exceção da porcentagem de clientes com janela de tempo, que é de 100% no problema R101, 75% no problema R102, 50% no problema R103 e 25% no problema

R104. Já os problemas R105 a R108 são idênticos aos problemas R101 a R104, aumentando apenas a amplitude da janela de tempo. O mesmo ocorre para os problemas R106 a R112.

Nesse conjunto de problemas, considera-se frota homogênea, cada cliente pode ser atendido uma única vez, por um único veículo (a demanda de cada cliente é menor que a capacidade do veículo), e o objetivo é minimizar a distância total percorrida e o tempo total das rotas, de forma que as restrições de capacidade dos veículos e janelas de tempo sejam respeitadas.

Nesse sentido, uma adaptação do algoritmo proposto no presente trabalho é, primeiramente, implementado no conjunto de problemas de Solomon (1987), com o objetivo de avaliar seu desempenho para problemas de roteirização de veículos com janelas de tempo. A seguir, está descrito cada um dos passos adaptados da metaheurística *scatter search* (SS) para as instâncias de Solomon (1987).

1. Método de geração de soluções diversas

O conjunto de soluções diversas de SS é gerado a partir da heurística de inserção seqüencial I1 de Solomon (1987). Durante as três primeiras iterações, as soluções são geradas incorporando o método GRACO1. Para as iterações restantes, aplica-se o método GRACO2.

2. Método de Melhoria

Como, em problemas de roteirização com janelas de tempo, cada cliente só pode ser atendido por um único veículo, as melhorias realocação de demanda e adição de rotas não são implementadas. São implementados, portanto, o método de trocas na mesma rota, eliminação de rotas e inserção.

No método de eliminação de rotas, ao invés de eliminar veículos menores que o segundo maior veículo, eliminam-se rotas com n ou menos clientes. Além disso, para cada possível rota eliminada, ao invés de combiná-la com uma única rota, cada cliente é inserido em uma nova rota, com o objetivo de não violar a restrição de capacidade do veículo. O algoritmo de eliminação de rotas pode ser expresso da seguinte forma:

Figura 6.8 - Algoritmo de eliminação e inserção para instâncias de Solomon (1987)

Para cada cliente i da possível rota R_j a ser eliminada (inicia-se pela rota com maior distância e pelo cliente mais distante da rota)

Escolher uma rota R_k entre as quatro primeiras rotas da lista candidata, sendo que a distância da rota R_j é calculada apenas em relação ao cliente i . Verificar se a rota R_k é viável, em termos de capacidade e tipo de veículo (após inserção do cliente i). Se a resposta for positiva, aplicar o próximo passo. Caso contrário, repetir o procedimento para as próximas quatro rotas da lista.

Inserir o cliente i da rota R_j em todas as posições da rota R_k e escolher a melhor posição viável (janela de tempo).

Se, após inserção de todos os clientes da rota R_j em novas rotas R'_k , houver melhoria (redução da distância), as inserções são efetuadas.

O algoritmo de inserção é semelhante ao descrito no item 6.5, porém, a viabilidade (antes do movimento de inserção) é testada apenas em relação à restrição de capacidade do veículo. Além disso, a melhoria é testada apenas em relação ao acréscimo ou decréscimo de distância, já que a frota de veículos é homogênea. O algoritmo completo de inserção pode ser definido como:

Figura 6.9 - Algoritmo de inserção para instâncias de Solomon (1987)

Para cada rota R_j (inicia-se com a rota mais longa)

Escolher uma rota, aleatoriamente, entre as quatro primeiras rotas R_k da lista candidata. Verificar se pelo menos uma loja da rota R_j é viável em termos de capacidade (após inserção na rota R_k). Se a resposta for positiva, passar para o passo seguinte. Caso contrário, aplicar o mesmo procedimento para as próximas quatro rotas da lista.

Inserir cada cliente viável da rota R_j (iniciar pelo cliente mais distante) em todas as posições da rota R_k e escolher a melhor delas, caso haja melhoria (distância) e a restrição de janela de tempo seja respeitada.

3. Método de criação e atualização do conjunto de referência

O método de criação e atualização do conjunto de referência não se altera.

4. Método de geração de subconjuntos

O método de combinação também é aplicado para todos os pares de soluções do conjunto de referência (combinações do tipo 1).

5. Método de combinação das soluções

Como o problema apresenta apenas um tipo de veículo e cada cliente só pode pertencer a uma rota, não será necessário determinar a quantidade entregue a cada cliente (é igual à demanda) e nem o tipo de veículo. A seguir, está descrita cada uma das fases do método de combinação.

FASE 1

Da mesma forma, inicia-se com a combinação dos componentes que possuem o maior número de elementos comuns. Porém, no caso de empate escolhe-se uma das combinações de rotas aleatoriamente. A rota combinada é formada pelos elementos comuns das rotas, sendo que a quantidade entregue (demanda) e o tipo de veículo já estão definidos. O roteiro da solução combinada é semelhante ao da melhor solução e, no caso de empate, escolhe-se o roteiro de uma das soluções aleatoriamente.

FASE 2

Esta fase tem como objetivo abastecer os clientes que ainda não foram atendidos, através de um procedimento de inserção. O cliente i a ser inserido é o cliente mais distante.

Com base nas soluções iniciais A e B , verifica-se a rota em que o cliente i está inserido, para cada uma das soluções, e todos os clientes j no qual $x_{ij} = 1$ ou $x_{ji} = 1$. Assim, para cada j pertencente a uma das rotas combinadas no passo A, com exceção do depósito

($j = 0$), calcula-se o custo de inserção do cliente i (acréscimo de distância e tempo), nas posições x_{ij} (quando $x_{ij} = 1$) e x_{ji} (quando $x_{ji} = 1$), e escolhe-se aquela com menor custo.

Para que a inserção seja efetuada, as restrições de capacidade dos veículos e janelas de tempo devem ser respeitadas. Caso nenhuma das inserções seja viável, adiciona-se uma nova rota.

Os demais passos são semelhantes aos descritos para o HFVRPTWSD.

6.6.2 INSTÂNCIAS DE PROBLEMAS DE HO E HAUGLAND (2004)

Ho e Haugland (2004) modificaram as demandas do conjunto de problemas de Solomon, de modo a permitir mais de uma entrega por cliente. Considera-se m a capacidade do veículo e w_i a demanda do cliente i ($i = 1, \dots, n$). Cada demanda é recalculada dentro do intervalo $[lm, um]$, onde $l < u$ estão definidos dentro do intervalo $[0, 1]$. Assim, para todo $i \in C$, define-se uma nova demanda $w'_i = lm + m((u - l)/(\bar{w} - \underline{w}))(w_i - \underline{w})$, onde $\underline{w} = \min\{w_i : i \in C\}$ e $\bar{w} = \max\{w_i : i \in C\}$. A nova demanda w'_i é arredondada para o inteiro mais próximo.

Para cada conjunto de problema os autores utilizam os seguintes valores de $[l, u]$:

Tabela 6.6 – Valores de demanda

<i>l</i>	<i>u</i>
0,01	0,50
0,02	1,00
0,50	1,00
0,70	1,00

Desta forma, foram gerados 224 problemas. Apesar das mudanças nos valores das demandas, verificou-se que, para todos os casos, a demanda de cada cliente nunca excede a capacidade do veículo, sendo no máximo igual à sua capacidade.

O VRPTWSD considera frota homogênea, cada cliente pode ser abastecido por mais de um veículo (a demanda de cada cliente é menor ou igual à capacidade do veículo) e o objetivo é minimizar a distância total percorrida e o tempo total das rotas, de forma que as restrições de capacidade dos veículos e janelas de tempo sejam respeitadas. A seguir são descritas as adaptações do algoritmo proposto no presente trabalho para o VRPTWSD.

1. Método de geração de soluções diversas

O conjunto P de soluções diversas é gerado a partir de uma adaptação da heurística AHH para frota homogênea, incorporando o método GRACO1 para as três primeiras iterações e o GRACO2 para as demais. Esta heurística também pode ser considerada uma adaptação da heurística de inserção I1 de Solomon (1987), porém, para que um cliente seja inserido em uma determinada rota, a restrição de capacidade do veículo não precisa ser respeitada, de forma que a demanda é adicionada enquanto houver capacidade.

Na heurística construtiva implementada, o critério de início também utiliza o cliente mais distante ainda não alocado. O critério de inserção busca minimizar o acréscimo de distância e tempo causados pela inserção de um cliente. Inicia-se, portanto, com o cliente i mais distante. A próxima etapa é a inserção de um novo cliente j na rota. Caso a demanda do cliente j , somada com a demanda do cliente i , exceda a capacidade do veículo, a demanda do cliente j é adicionada enquanto houver capacidade e a demanda remanescente é adicionada a um novo veículo. Este processo continua até que todos os clientes pertençam a um roteiro.

2. Método de Melhoria

Todas as melhorias implementadas para o HFVRPTWSD também são implementadas para este problema. Os movimentos de troca na mesma rota, realocação de demanda e adição de rotas são semelhantes aos descritos no item 6.5, porém, baseados em um único veículo.

O método de eliminação de rotas e o método de inserção são semelhantes aos descritos no item 6.6.1 para as instâncias de Solomon (1987).

3. Método de criação e atualização do conjunto de referência

O método de criação e atualização do conjunto de referência não se altera.

4. Método de geração de subconjuntos

O método de combinação também é aplicado para todos os pares de soluções do conjunto de referência (combinações do tipo 1).

5. Método de combinação das soluções

A fase 1 do método de combinação é semelhante ao algoritmo proposto para HFVRPTWSD, porém, baseado em um único tipo veículo. A fase 2 tem como objetivo atender a demanda remanescente dos clientes, através de um procedimento de inserção. O cliente i a ser inserido é o cliente mais distante. As adaptações dos passos A e B da fase 2 estão descritas a seguir.

PASSO A

Caso o cliente i já pertença a pelo menos uma rota combinada (fase 1 do método de combinação), escolhe-se aquela com maior capacidade ociosa e entrega-se o mínimo entre a capacidade ociosa do veículo e a demanda do cliente i . Enquanto a demanda total do cliente i não for atendida e houver uma rota com capacidade ociosa em que o cliente i está inserido, o procedimento é repetido.

Ao final deste passo, caso a demanda total do cliente i ainda não tenha sido atendida, aplica-se o passo B. Caso contrário, escolhe-se o próximo cliente mais distante a ser inserido.

PASSO B

Com base nas soluções iniciais A e B , verificam-se todas as rotas em que o cliente i está inserido e todos os arcos no qual $x_{ij} = 1$ ou $x_{ji} = 1$. Assim, para cada j pertencente a uma

das rotas combinadas no passo A, com exceção do depósito ($j = 0$), calcula-se o custo de inserção do cliente i (acréscimo de distância e tempo) nas posições x_{ij} (quando $x_{ij} = 1$) e x_{ji} (quando $x_{ji} = 1$). Considera-se $x_{i0} = 1$ ou $x_{0i} = 1$ apenas para rotas com carga completa ($0 - i - 0$).

Para cada possível posição de inserção do cliente i , entrega-se o mínimo, entre a capacidade ociosa do veículo e a demanda do cliente i , e escolhe-se aquela com menor custo, desde que as restrições de janelas de tempo sejam respeitadas. O procedimento é repetido até que a demanda total do cliente i seja atendida ou enquanto houver uma posição de inserção do cliente i . Verificadas todas as rotas das soluções A e B em que o cliente i está inserido, caso a demanda total não tenha sido atendida, adiciona-se uma nova rota. O procedimento é repetido até que a demanda total do cliente i seja atendida.

Os demais passos são semelhantes aos descritos para o problema estudado.

6.6.3 INSTÂNCIAS DE PROBLEMAS DE LIU E SHEN (1999)

Liu e Shen (1999) adaptaram o conjunto de problemas de Solomon (1987) para frota heterogênea, gerando um total de 168 problemas que estão detalhados em anexo (ANEXO C). Nesse novo conjunto de problemas, cada cliente pode ser atendido por apenas um veículo (a quantidade entregue é igual à demanda). O objetivo é minimizar a soma dos custos fixos e de roteirização (distância), além do tempo total das rotas, respeitando as restrições de capacidade dos veículos e janelas de tempo.

1. Método de geração de soluções diversas

O conjunto P de soluções diversas é gerado a partir da heurística construtiva de Dullaert et al. (2002), incorporando o método GRACO1 nas três primeiras iterações e o método GRACO2 nas demais iterações.

2. Método de Melhoria

Como, em problemas de dimensionamento e roteirização de uma frota heterogênea com janelas de tempo, cada cliente pode ser abastecido por apenas um veículo, as melhorias realocação de demanda e adição de rotas não são implementadas. São implementados, portanto, o método de trocas na mesma rota, eliminação e combinação de rotas e inserção.

Os movimentos de eliminação e combinação de rotas e inserção são semelhantes aos descritos para o HFVRPTWSD, porém, a restrição quanto ao de tipo de veículo não é considerada.

3. Método de criação e atualização do conjunto de referência

O método de criação e atualização do conjunto de referência não se altera.

4. Método de geração de subconjuntos

O método de combinação também é aplicado para todos os pares de soluções do conjunto de referência (combinações do tipo 1).

5. Método de combinação das soluções

FASE 1

Da mesma forma que no HFVRPTWSD, inicia-se com a combinação dos componentes que possuem o maior número de elementos comuns. Porém, no caso de empate, escolhe-se uma das combinações de rotas aleatoriamente. A rota combinada será formada pelos elementos comuns das rotas, sendo que a quantidade entregue (demanda) já está definida. O veículo designado é o menor capaz de atender a demanda total da rota. O roteiro da solução combinada é semelhante ao da melhor solução e, no caso de empate, escolhe-se o roteiro de uma das soluções aleatoriamente.

FASE 2

Esta fase tem como objetivo abastecer os clientes restantes que não foram atendidos na fase 1, através de um procedimento de inserção. O cliente i a ser inserido é o cliente mais

distante. Com base nas soluções iniciais A e B , verifica-se a rota em que o cliente i está inserido, em cada uma das soluções, e os clientes j no qual $x_{ij} = 1$ ou $x_{ji} = 1$. Assim, para cada j pertencente a uma das rotas combinadas no passo A, com exceção do depósito ($j = 0$), calcula-se o custo de inserção do cliente i (acrédito de custo fixo, distância e tempo), para as posições x_{ij} (quando $x_{ij} = 1$) e x_{ji} (quando $x_{ji} = 1$), e escolhe-se aquela com menor custo. Considera-se $x_{i0} = 1$ ou $x_{0i} = 1$ apenas para rotas com carga completa ($0 - i - 0$).

Para que a inserção seja efetuada, as restrições de capacidade dos veículos e janelas de tempo devem ser respeitadas. Caso nenhuma das inserções de um determinado cliente i seja viável, adiciona-se uma nova rota.

Os demais passos são semelhantes.

CAPÍTULO 7

ANÁLISE DOS RESULTADOS

Os métodos descritos no capítulo anterior foram, primeiramente, aplicados para resolução de um caso real de abastecimento de lojas de um grupo varejista brasileiro. Posteriormente, os algoritmos foram adaptados para resolver as seguintes instâncias de problemas da literatura: Solomon (1987), Ho e Haugland (2004) e Liu e Shen (1999). Os resultados do estudo de caso, do conjunto de problemas de Solomon (1987), de Ho e Haugland (2004) e de Liu e Shen (1999) encontram-se nos itens 7.1 a 7.4, respectivamente.

7.1 ESTUDO DE CASO

Os algoritmos foram aplicados para resolução de um problema real de distribuição, considerando um dia típico de operação. Para que os resultados possam ser comparados, o modelo de valor de frete foi aplicado à solução obtida pela empresa. Foi escolhida uma data significativa que representa a média das operações no mês. Além desta data, foram analisados outros 6 dias na mesma semana, com o objetivo de verificar o impacto na qualidade das soluções decorrente da mudança no volume. O custo total de abastecimento, para cada uma das heurísticas construtivas e para a metaheurística *scatter search* (SS), considerando os 7 dias da semana, está na tabela 7.1 e no gráfico 7.1.

Tabela 7.1 – Custo total das soluções

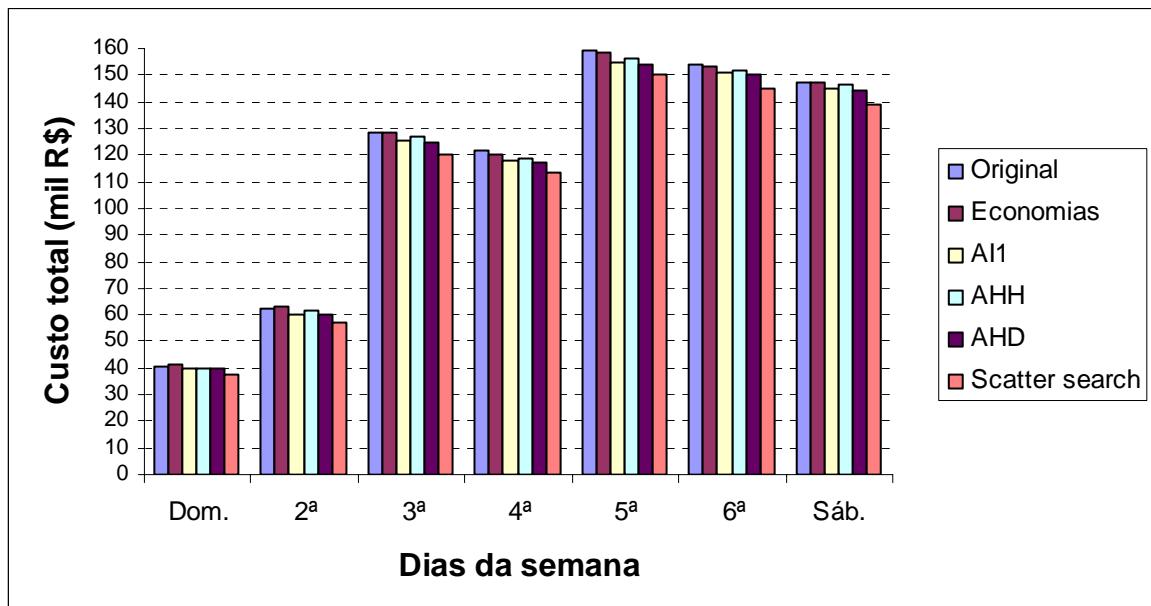
Método	Custo Total (R\$)						
	Dom.	2^a	3^a	4^a	5^a	6^a	Sáb.
Original	40.725	62.112	128.724	121.534	159.289	154.248	147.372
Economias	41.163	62.987	128.078	120.142	158.746	153.416	147.236
AI1	39.854	60.014	125.412	117.985	154.769	150.696	145.033
AHH	40.141	61.751	126.864	118.597	156.141	151.856	146.213
AHD	39.571	59.751	124.842	117.125	154.257	150.022	144.539
Scatter search	37.427	57.034	120.041	113.717	150.338	145.347	139.015

Legenda:

AI1: Adaptação da heurística de inserção seqüencial de Solomon (1987)

AHH: Adaptação da heurística de inserção seqüencial de Ho e Haugland (2004)

AHD: Adaptação da heurística de inserção seqüencial de Dullaert et al. (2002)

Gráfico 7.1 – Custo total das soluções

Conforme mostram o gráfico 7.1 e a tabela 7.1, a heurística construtiva que apresentou o melhor resultado foi AHD (adaptação da heurística de inserção seqüencial de Dullaert et al.). A heurística AI1 (adaptação da heurística de inserção seqüencial de Solomon) apresentou resultados bastante similares, porém, um pouco inferiores. A heurística AHH (adaptação da heurística de inserção seqüencial de Ho e Haugland) apresentou resultados inferiores às heurísticas AHD e AI1. Os resultados do algoritmo de economias foram inferiores às demais heurísticas.

A economia obtida durante a semana, para a metaheurística *scatter search*, foi em torno de R\$51.000. Se projetarmos este resultado para os demais dias do ano, chegaremos a uma economia em torno de R\$2.6 milhões por ano.

Cabe ressaltar que, com base no histórico das entregas efetuadas pela empresa, muitas vezes a capacidade dos caminhões é violada. Isto ocorre pela possibilidade de remontagem de alguns paletes, de modo que o volume ocupado torna-se menor. Este procedimento não foi incorporado ao modelo, o que poderia gerar ganhos ainda maiores.

As diferenças percentuais em relação à solução original, para cada dia da semana, estão na tabela 7.2 e no gráfico 7.2.

Tabela 7.2 – Diferença percentual em relação à solução original

Método	Dom.	2 ^a	3 ^a	4 ^a	5 ^a	6 ^a	Sáb.
Economias	1,08%	1,41%	-0,50%	-1,15%	-0,34%	-0,54%	-0,09%
AI1	-2,14%	-3,38%	-2,57%	-2,92%	-2,84%	-2,30%	-1,59%
AHH	-1,43%	-0,58%	-1,44%	-2,42%	-1,98%	-1,55%	-0,79%
AHD	-2,83%	-3,80%	-3,02%	-3,63%	-3,16%	-2,74%	-1,92%
Scatter search	-8,10%	-8,18%	-6,75%	-6,43%	-5,62%	-5,77%	-5,67%

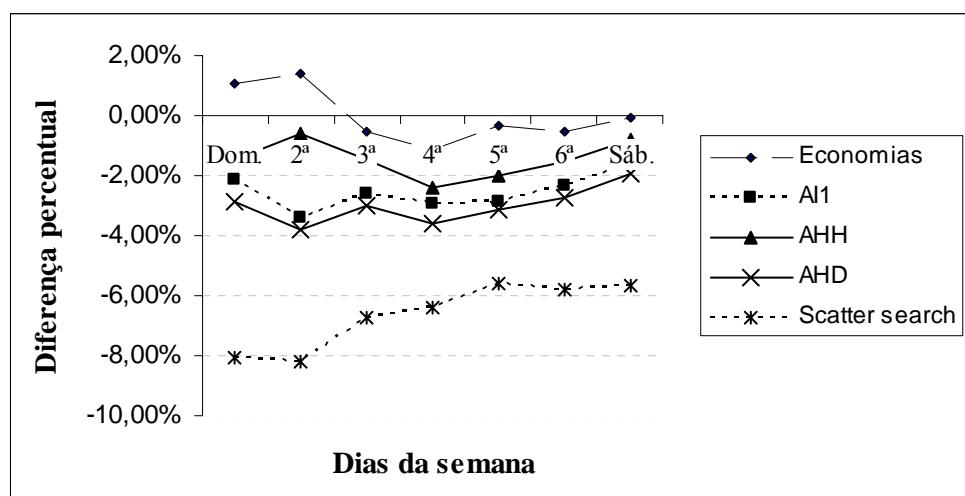
Legenda:

AI1: Adaptação da heurística de inserção seqüencial de Solomon (1987)

AHH: Adaptação da heurística de inserção seqüencial de Ho e Haugland (2004)

AHD: Adaptação da heurística de inserção seqüencial de Dullaert et al. (2002)

Gráfico 7.2 – Diferença percentual dos algoritmos em relação à solução original



Em termos de número de veículos utilizados, os resultados encontram-se a seguir.

Tabela 7.3 – Número de veículos utilizados

Método	Dom.	2 ^a	3 ^a	4 ^a	5 ^a	6 ^a	Sáb.
Original	118	145	290	274	365	359	343
Economias	120	147	289	273	369	361	342
AI1	111	136	290	264	361	350	335
AHH	98	117	258	229	322	312	304
AHD	115	141	288	257	364	354	320
Scatter search	110	140	289	258	358	349	329

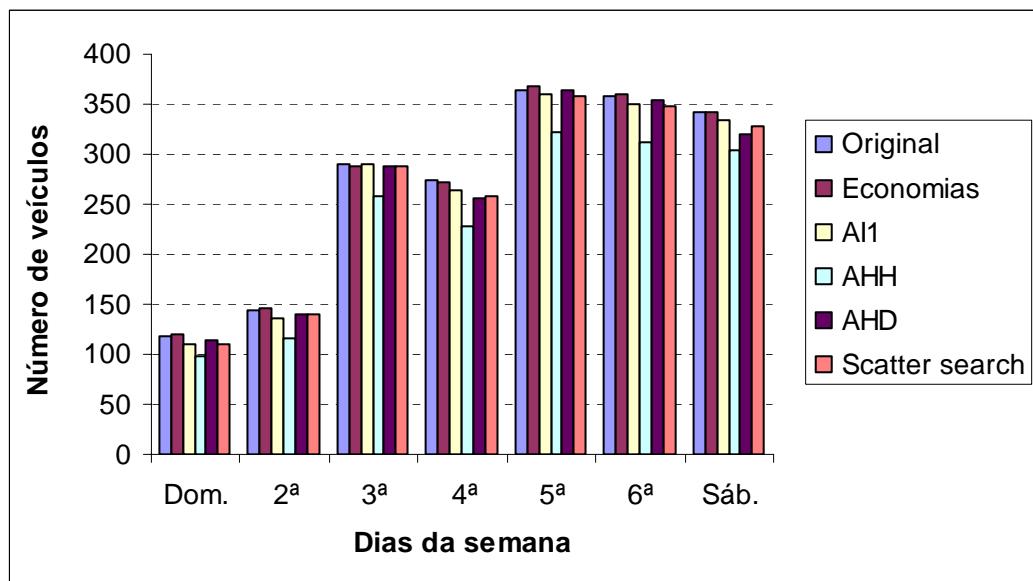
Legenda:

AI1: Adaptação da heurística de inserção seqüencial de Solomon (1987)

AHH: Adaptação da heurística de inserção seqüencial de Ho e Haugland (2004)

AHD: Adaptação da heurística de inserção seqüencial de Dullaert et al. (2002)

Gráfico 7.3 – Número de veículos utilizados



De acordo com a tabela 7.3 e o gráfico 7.3, podemos notar que a heurística de economias utiliza o maior número de veículos. Os resultados da heurística de economias são similares à solução original. A heurística de inserção seqüencial de Ho e Haugland adaptada (AHH) utiliza um número menor de roteiros em relação aos demais algoritmos, já que as demandas dos clientes podem ser quebradas e adicionadas aos veículos enquanto houver capacidade. Em função disto, a AHH foi utilizada como uma das soluções iniciais de *scatter*

search. As heurísticas AI1 (adaptação da heurística de inserção seqüencial de Solomon), AHD (adaptação da heurística de inserção seqüencial de Dullaert) e a metaheurística *scatter search* apresentaram resultados similares em termos de número de veículos.

A tabela 7.4 apresenta o tempo de processamento para cada um dos algoritmos e dias da semana.

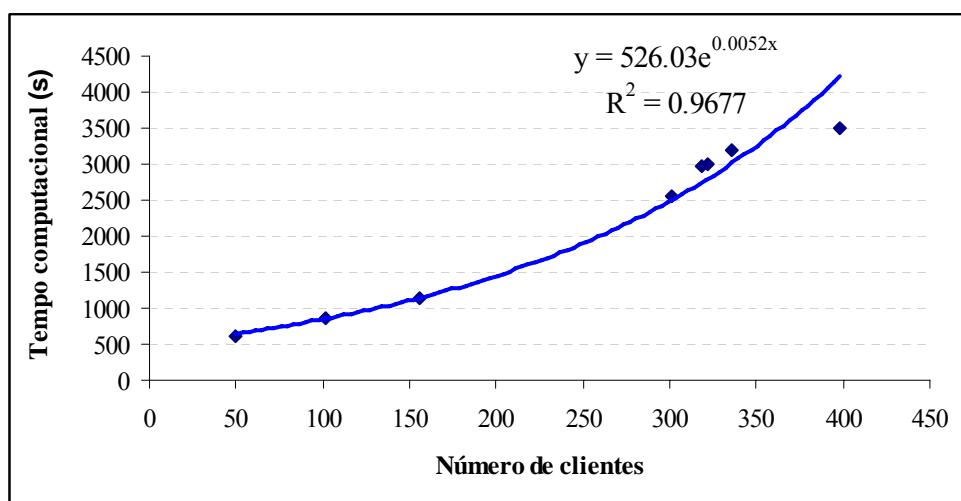
Tabela 7.4 – Tempo de processamento

Algoritmo	Tempo (s)						
	Dom.	2^a	3^a	4^a	5^a	6^a	Sáb.
Economias	9	10	17	14	20	18	17
AI1	20	22	38	36	44	40	38
AHH	18	19	33	30	39	35	32
AHD	22	24	42	37	49	44	40
Scatter search	860	1148	3092	3004	3494	3198	2986

O método de economias apresentou o menor tempo de processamento em todos os casos, porém, os resultados não são satisfatórios.

O gráfico 7.4 apresenta o tempo computacional da metaheurística *scatter search*, em segundos, para problemas com $n = 50, 100, 150, 200, 250, 300, 350, 400$ clientes.

Gráfico 7.4 – Tempo computacional da metaheurística *scatter search*



A seguir, são mostrados os resultados detalhados da metaheurística *scatter search*, para um dia típico de operação (4^a feira) que representa a média das operações no mês. Os dados de saída (roteiros de entrega) para este dia encontram-se no Apêndice A.

Conforme apresentado no capítulo anterior, em função dos resultados obtidos das heurísticas construtivas, no método de diversificação foram considerados os seguintes casos:

- a) Caso 1: 100% das soluções diversas são geradas a partir da heurística de inserção AHD.
- b) Caso 2: 100% das soluções diversas são geradas a partir da heurística de inserção AHH.
- c) Caso 3: 50% das soluções diversas são geradas a partir da heurística de inserção AHD e as 50% restantes a partir da heurística de inserção AHH.

Os demais parâmetros considerados foram: $PSize = 30$, $b_1 = 5$, $b_2 = 5$, atualização dinâmica e por qualidade e diversidade. Os resultados, para cada um dos casos, encontram-se na tabela 7.5.

Tabela 7.5 – Método de diversificação

	Caso 1	Caso 2	Caso 3
Custo (R\$)	114.308	115.173	113.717

De acordo com a tabela acima, o melhor resultado obtido foi para o caso 3. Para o caso 1, os resultados foram um pouco inferiores. Já o caso 2 obteve o pior resultado.

Para diferentes valores de $PSize$ (10, 20, 30, 40), fixando os demais parâmetros (método de diversificação gerado através do caso 3, $b_1 = 5$, $b_2 = 5$, atualização dinâmica e por qualidade e diversidade), foram obtidos os seguintes resultados:

Tabela 7.6 – Resultados para diferentes valores de $PSize$

$PSize$	10	20	30	40
Custo (R\$)	116.265	114.618	113.717	113.685
Tempo (s)	1021	2187	3004	4133

De acordo com a tabela 7.6, podemos verificar que o melhor resultado obtido foi $PSize = 30$. Para $PSize = 40$, o custo foi um pouco inferior, porém, o tempo computacional foi significativamente superior. Desta forma foi escolhido $PSize = 30$.

Portanto, para os demais testes do método *scatter search*, são utilizados os seguintes parâmetros: tamanho do $PSize = 30$, método de diversificação gerado através do caso 3 (50% das soluções foram geradas pela heurística AHD e as 50% soluções restantes pela heurística AHH, incorporando o método GRACO1 para as 3 primeiras iterações e GRACO2 para as iterações restantes). O método de melhoria (trocas na mesma rota, realocação de demanda, combinação e eliminação de rotas, inserção e adição de rotas), o método de geração de subconjuntos (dois a dois), o método de combinação e o critério de parada (5 iterações com um limite de tempo de 1 hora) não se alteram. Para os demais parâmetros, foram testados os seguintes valores:

- Valores do conjunto de referência (b_1, b_2): (3,2), (5,5), (7,3) e (3,7)
- Critérios de atualização do conjunto de referência: QD ou Q
- Freqüência de atualização do conjunto de referência: Estática ou Dinâmica

Os resultados encontram-se na tabela 7.7.

Tabela 7.7 – Resultados de *scatter search* para diferentes parâmetros

Parâmetros		Custo	
$b_1 = 3$	QD	Estático	115.088
		Dinâmico	115.323
	Q	Estático	115.064
		Dinâmico	114.927
$b_1 = 5$	QD	Estático	114.025
		Dinâmico	113.717
	Q	Estático	114.443
		Dinâmico	114.274
$b_1 = 7$	QD	Estático	114.185
		Dinâmico	114.410
	Q	Estático	114.350
		Dinâmico	114.175
$b_1 = 3$	QD	Estático	114.501
		Dinâmico	114.466
	Q	Estático	114.598
		Dinâmico	114.683

Dentre as versões do método *scatter search* testadas, o melhor resultado obtido foi para $b_1 = 5$, $b_2 = 5$, critério de atualização por qualidade e diversidade (QD) e freqüência de atualização dinâmica.

7.2 INSTÂNCIAS DE SOLOMON (1987)

Conforme descrito no item 6.6.1, o algoritmo proposto foi adaptado para resolver o conjunto de problemas de Solomon (1987). Primeiramente, foram testados diferentes valores de b_1 e b_2 . Os demais parâmetros considerados foram: $PSize = 30$, critério de atualização por qualidade (Q) e freqüência de atualização dinâmica. Os valores médios, para cada uma das classes de problemas de Solomon, encontram-se na tabela 7.8.

Tabela 7.8 – Resultados de scatter search para diferentes valores de b_1, b_2 .

Problema	$b_1 = 3, b_2 = 2$		$b_1 = 5, b_2 = 5$		$b_1 = 7, b_2 = 3$		$b_1 = 3, b_2 = 7$	
	Distância	Veículos	Distância	Veículos	Distância	Veículos	Distância	Veículos
R1	1252.91	12.58	1230.52	12.33	1234.01	12.42	1223.58	12.33
C1	855.04	10.00	837.60	10.00	839.66	10.00	834.98	10.00
RC1	1412.20	12.13	1377.45	11.88	1378.24	11.88	1388.32	11.88
R2	992.90	2.91	971.74	2.82	975.03	2.82	964.97	2.82
C2	615.58	3.00	599.53	3.00	598.85	3.00	596.13	3.00
RC2	1169.32	3.75	1145.11	3.50	1148.04	3.50	1138.84	3.50
Total	6297.95	44.37	6161.95	43.53	6173.83	43.62	6146.82	43.53
Tempo^a		221		612		624		638

^a Tempo médio de processamento (segundos).

De acordo com a tabela 7.8, o melhor resultado obtido foi para $b_1 = 3$ e $b_2 = 7$. No caso de $b_1 = 3$ e $b_2 = 2$, obteve-se o menor tempo computacional, porém os resultados foram inferiores. O tempo de processamento (segundos) representa a média dos 56 problemas de Solomon.

Considerando os demais parâmetros, o melhor resultado obtido foi para $PSize = 30$, critério de atualização por qualidade (Q) e freqüência de atualização dinâmica.

A tabela 7.9 compara os resultados, em termos de distância e número de veículos, do algoritmo apresentado neste trabalho com os melhores resultados encontrados na literatura, para cada uma das instâncias de problemas de Solomon (1987).

Tabela 7.9 – Resultados de *scatter search* para as instâncias de Solomon (1987)

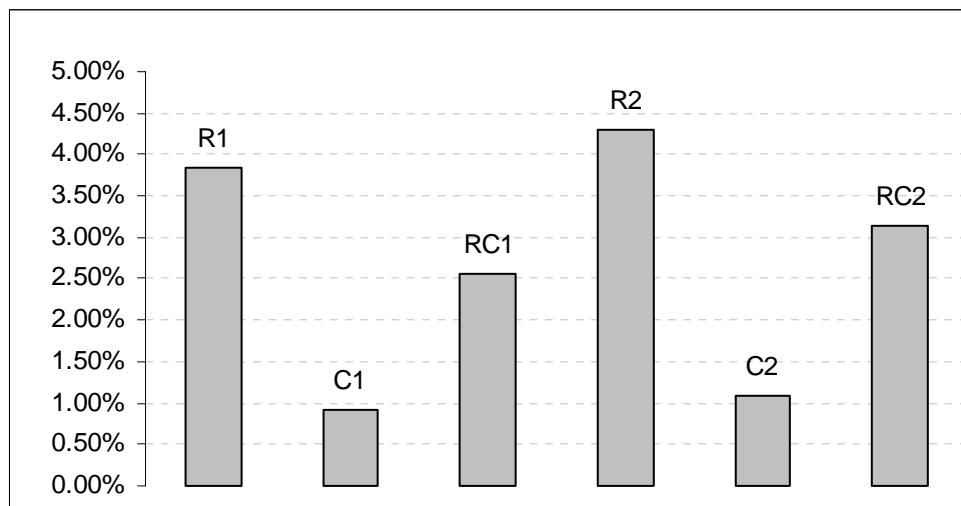
Problema	Melhor resultado		<i>Scatter search</i>		
	Distância	Veículos	Distância	Veículos	Diferença %
R101	1607.70	18	1651.84	19	-
R102	1434.00	17	1495.83	17	4.31
R103	1292.68	13	1302.41	14	-
R104	1007.31	9	989.32	10	-
R105	1377.11	14	1450.27	14	5.31
R106	1251.98	12	1283.75	12	2.54
R107	1104.66	10	1134.96	10	2.74
R108	960.88	9	993.58	9	3.40
R109	1194.73	11	1161.93	12	-
R110	1118.59	10	1172.47	10	4.82
R111	1096.72	10	1076.48	11	-
R112	982.14	9	970.06	10	-
média	1202.37	11.83	1223.58	12.33	3.85
C101	827.30	10	828.94	10	0.20
C102	827.30	10	838.90	10	1.40
C103	826.30	10	832.48	10	0.75
C104	822.90	10	840.65	10	2.16
C105	827.30	10	841.34	10	1.70
C106	827.30	10	839.63	10	1.49
C107	827.30	10	828.94	10	0.20
C108	827.30	10	828.94	10	0.20
C109	827.30	10	828.94	10	0.20
média	826.70	10	834.31	10	0.92
RC101	1696.94	14	1640.39	15	-
RC102	1554.75	12	1491.12	13	-
RC103	1261.67	11	1307.88	11	3.66
RC104	1135.48	10	1170.81	10	3.11
RC105	1629.44	13	1678.41	13	3.01
RC106	1424.73	11	1409.81	12	-
RC107	1230.48	11	1268.34	11	3.08
RC108	1139.82	10	1139.82	10	0
média	1325.52	11.88	1388.32	11.88	2.57

Continuação da Tabela 7.9 – Resultados de *scatter search* para as instâncias de Solomon (1987)

Problema	Melhor resultado			<i>Scatter search</i>		
	Distância	Veículos	Distância	Veículos	Diferença %	
R201	1252.37	4	1283.41	4	2.48	
R202	1191.70	3	1237.12	3	3.81	
R203	939.54	3	980.62	3	4.37	
R204	825.52	2	855.79	2	3.67	
R205	994.42	3	1045.69	3	5.16	
R206	906.14	3	958.24	3	5.75	
R207	890.61	2	937.40	2	5.25	
R208	726.75	2	756.99	2	1.41	
R209	909.16	3	954.77	3	5.02	
R210	939.34	3	971.73	3	3.45	
R211	892.71	2	833.45	3	-	
média	951.66	2.73	964.97	2.82	4.31	
C201	591.56	3	599.37	3	1.32	
C202	591.56	3	591.56	3	0	
C203	591.17	3	591.17	3	0	
C204	590.60	3	614.55	3	4.06	
C205	588.88	3	599.08	3	1.73	
C206	588.49	3	597.32	3	1.50	
C207	588.29	3	588.29	3	0	
C208	588.32	3	588.32	3	0	
média	589.86	3	596.13	3	1.08	
RC201	1406.91	4	1454.62	4	3.39	
RC202	1365.65	3	1370.71	3	0.37	
RC203	1049.62	3	1053.96	3	0.41	
RC204	798.41	3	833.89	3	4.44	
RC205	1297.19	4	1337.81	4	3.13	
RC206	1146.32	3	1127.44	4	-	
RC207	1061.14	3	1045.55	4	-	
RC208	828.14	3	886.75	3	7.08	
média	1119.17	3.25	1138.84	3.5	3.14	

Fonte: Marius M. Solomon <http://web.cba.neu.edu/~msolomon/problems.htm>.

A diferença média percentual, para as classes R1, C1, RC1, R2, C2 e RC2, dos melhores resultados obtidos da literatura em relação à metaheurística *scatter search*, está no gráfico 7.5.

Gráfico 7.5 – Diferença média percentual para as classes de Solomon

De acordo com a tabela 7.9, podemos notar que a metaheurística *scatter search* alcançou os melhores resultados encontrados na literatura em 5 problemas. Além disso, em outros 10 problemas, o algoritmo reduziu a distância total percorrida, porém, foi necessária a utilização de mais um veículo. De acordo com o gráfico 7.5, verificou-se que, em média, os problemas R1, C1 e RC1 apresentaram melhores resultados que as classes R2, C2 e RC2. Além disso, os problemas da classe C apresentaram, em média, melhores resultados que os problemas da classe R e RC. O pior resultado foi apresentado pelos problemas da classe R. Podemos concluir, portanto, que o algoritmo se adapta melhor a horizontes curtos, rotas com poucos clientes, pequenos veículos e clientes agrupados e semi-agrupados geograficamente.

A tabela 7.10 apresenta o tempo médio de processamento (segundos) para cada uma das classes de problemas de Solomon (1987).

Tabela 7.10 - Tempo médio de processamento (segundos) para as classes de problemas de Solomon (1987)

R1	C1	RC1	R2	C2	RC2
503	576	587	730	662	771

7.3 INSTÂNCIAS DE HO E HAUGLAND (2004)

Da mesma forma que no item 7.2, primeiramente, foram testados diferentes valores de b_1 e b_2 . Os demais parâmetros considerados foram: $PSize = 20$, critério de atualização por qualidade e diversidade (QD) e freqüência de atualização estática. Estes parâmetros foram testados para valores de $[l, u] = [0.70, 1.00]$. Os valores médios, para as classes R1, C1, RC1, R2, C2 e RC2, encontram-se na tabela 7.11.

Tabela 7.11 – Resultados de *scatter search* para diferentes valores de b_1, b_2 , considerando $[l, u] = [0.70, 1.00]$.

Problema	$b_1 = 3, b_2 = 2$		$b_1 = 5, b_2 = 5$		$b_1 = 7, b_2 = 3$		$b_1 = 3, b_2 = 7$	
	Distância	Veículos	Distância	Veículos	Distância	Veículos	Distância	Veículos
R1	4485.11	83.42	4464.85	82.75	4471.42	82.75	4470.98	83.42
C1	4961.41	77.50	4950.81	76.88	4954.12	76.88	4952.33	76.88
RC1	6047.80	82.50	6013.92	82.50	6012.74	82.50	6011.45	82.50
R2	4665.96	83.18	4625.87	82.82	4629.30	82.82	4630.81	83.00
C2	5229.33	77.50	5214.79	76.88	5213.96	77.50	5217.20	76.88
RC2	6237.58	83.88	6217.43	83.25	6220.60	83.25	6219.52	83.25
Total	31627.19	487.98	31487.67	485.08	31502.14	485.70	31502.29	485.93
Tempo^a		302		711		715		715

^a Tempo médio de processamento (segundos).

De acordo com a tabela 7.11, o melhor resultado obtido foi para $b_1 = 5$ e $b_2 = 5$. Da mesma forma que no item 7.2, no caso de $b_1 = 3$ e $b_2 = 2$, obteve-se o menor tempo computacional, porém os resultados foram inferiores. O tempo computacional (segundos) representa a média dos 56 problemas.

Considerando os demais parâmetros, o melhor resultado foi obtido para $PSize = 20$, critério de atualização por qualidade e diversidade (QD) e freqüência de atualização estática.

A tabela 7.12 compara os resultados do algoritmo apresentado neste trabalho com os melhores resultados encontrados na literatura (HO; HAUGLAND, 2004), para cada classe de problemas, considerando diferentes valores de $[l, u]$.

Tabela 7.12 – Comparação com os resultados de Ho e Haugland (2004) para as demandas modificadas de Solomon

<i>l, u</i>		R1	C1	RC1
0.01, 0.50	VRPTWSD ^a	18.25/1471.49	12.22/1182.12	20.13/1965.05
	VRPTWSD ^b	18.42/1475.54	12.22/1160.74	21.00/1941.25
0.02, 1.00	VRPTWSD ^a	35.00/2291.46	22.22/2168.57	40.00/3339.20
	VRPTWSD ^b	35.83/2302.58	24.00/2009.37	41.75/3425.96
0.50, 1.00	VRPTWSD ^a	67.00/4040.67	61.00/3979.78	70.00/5453.10
	VRPTWSD ^b	69.50/4035.84	60.75/3975.49	73.75/5231.85
0.70, 1.00	VRPTWSD ^a	79.00/4581.54	77.00/4962.28	81.00/6095.20
	VRPTWSD ^b	82.75/4464.85	76.88/4950.81	82.50/6013.92
<i>l, u</i>		R2	C2	RC2
0.01, 0.50	VRPTWSD ^a	18.00/1430.62	11.13/1174.29	20.00/1946.07
	VRPTWSD ^b	18.00/1425.40	11.75/1180.34	21.00/1941.42
0.02, 1.00	VRPTWSD ^a	35.00/2318.04	22.00/1995.59	39.00/3419.85
	VRPTWSD ^b	35.82/2314.65	23.13/1993.47	41.50/3410.65
0.50, 1.00	VRPTWSD ^a	68.00/4059.26	61.00/4268.02	71.00/5546.20
	VRPTWSD ^b	69.27/4055.29	60.88/4259.14	71.00/5498.32
0.70, 1.00	VRPTWSD ^a	80.00/4574.17	77.00/5246.12	82.00/6155.49
	VRPTWSD ^b	82.82/4625.87	76.88/5214.79	83.25/6217.43

Legenda

VRPTWSD^a: Resultados do VRPTWSD de Ho e Haugland (2004)

VRPTWSD^b: Resultados do VRPTWSD do presente trabalho

De acordo com a tabela 7.12, podemos concluir que, em 6 classes de problemas, a média da metaheurística *scatter search* superou os melhores resultados obtidos por Ho e Haugland (2004). Além disso, em outros 11 problemas, foi possível reduzir a distância média percorrida, porém, a média de veículos utilizados foi maior. Verificou-se, também, que os melhores resultados foram encontrados para valores de $[l,u] = [0.50,1.00]$ e $[0.70,1.00]$, pois, com maiores valores de demanda, aumenta a possibilidade de ocorrência de entregas fracionadas.

A tabela 7.13 mostra o tempo médio de processamento (segundos), para as classes de problemas R1, C1, RC1, R2, C2 e RC2, considerando diferentes valores de $[l, u]$.

Tabela 7.13 - Tempo médio de processamento (segundos) para as demandas modificadas de Ho e Haugland (2004)

l, u	R1	R2	C1	C2	RC1	RC2
0.01, 0.50	807	982	517	1121	678	1030
0.02, 1.00	840	1022	785	888	905	926
0.50, 1.00	1025	1014	574	876	1014	885
0.70, 1.00	812	899	755	922	901	957

7.4 INSTÂNCIAS DE LIU E SHEN (1999)

Os algoritmos também foram aplicados para resolução do conjunto de problemas de Liu e Shen (1999). Da mesma forma que nos itens 7.1 e 7.2, primeiramente, foram testados diferentes valores de b_1 e b_2 . Os demais parâmetros considerados foram: $PSize = 30$, critério de atualização por qualidade e diversidade (QD) e freqüência de atualização dinâmica. Estes parâmetros foram testados para o conjunto de problemas do tipo A. Os valores médios (tempo total de viagem excluindo o tempo de serviço, ou seja, soma dos tempos de percurso e de espera), para as classes $R1_A$, $C1_A$, $RC1_A$, $R2_A$, $C2_A$ e $RC2_A$, considerando diferentes valores de b_1 e b_2 , encontram-se na tabela 7.14.

Tabela 7.14 – Resultados de *scatter search* para diferentes valores de b_1, b_2 , considerando o conjunto de problemas do tipo A.

Problema	$b_1 = 3, b_2 = 2$	$b_1 = 5, b_2 = 5$	$b_1 = 7, b_2 = 3$	$b_1 = 3, b_2 = 7$
R1_A	1525	1520	1514	1521
C1_A	1180	1169	1153	1162
RC1_A	1660	1648	1637	1645
R2_A	1435	1425	1407	1421
C2_A	841	836	831	835
RC2_A	1782	1776	1765	1771
Total	8423	8374	8307	8355
Tempo^a	253	684	678	680

^a Tempo médio de processamento (segundos).

De acordo com a tabela 7.14, o melhor resultado obtido foi para $b_1 = 7$ e $b_2 = 3$. O tempo computacional (segundos) representa a média dos 56 problemas do tipo A.

Considerando os demais parâmetros, o melhor resultado foi obtido para $PSize = 30$, critério de atualização por qualidade e diversidade (QD) e freqüência de atualização dinâmica.

A tabela 7.15 compara os resultados da metaheurística *scatter search* com os melhores resultados encontrados na literatura, para cada classe de problemas.

Tabela 7.15 – Comparaçāo de *scatter search* com os resultados de Liu e Shen (1999) e Dullaert et al. (2002)

	Liu e Shen (1999)			Dullaert (2002)			<i>Scatter search</i>	
	MCS	MOOS	MROS	ACS	AOOS	AROS	AOOS	Δ %
R1_A	4398	4401	4403	1665	1549	1556	1514	-2,25
R1_B	2066	2054	2068	1617	1575	1557	1532	-1,60
R1_C	1716	1700	1706	1689	1577	1558	1528	-1,93
C1_A	8007	8295	8007	1248	1248	1166	1153	-1,11
C1_B	2661	2485	2661	1164	1126	1131	1099	-2,39
C1_C	1749	1705	1749	1435	1283	1155	1141	-1,21
RC1_A	5262	5184	5198	1778	1687	1665	1637	-1,68
RC1_B	2253	2252	2235	1781	1697	1681	1650	-1,84
RC1_C	1853	1859	1849	1887	1745	1690	1675	-0,88
R2_A	3809	3975	3809	1444	1435	1427	1407	-1,40
R2_B	1816	1797	1816	1457	1431	1446	1401	-2,09
R2_C	1513	1530	1513	1439	1420	1445	1392	-1,97
C2_A	6717	6889	6717	821	1072	1043	831	1,22
C2_B	1978	1970	1978	821	932	1043	821	0
C2_C	1288	1300	1288	811	828	1029	800	-1,36
RC2_A	5324	5273	5324	1802	1801	1805	1765	-2,00
RC2_B	2339	2338	2324	1742	1784	1770	1746	-1,36
RC2_C	1994	1978	1988	1754	1741	1962	1705	-2,06

De acordo com a tabela 7.15, podemos concluir que, para todas as classes de problemas de Liu e Shen (1999), com exceção das classes C2_A e C2_B, a metaheurística *scatter search* superou os melhores resultados encontrados na literatura.

O gráfico 7.6 apresenta a diferença média percentual, para as classes R1_A, C1_A,

R_{1A} , R_{2A} , C_{1A} e R_{C2A} , dos melhores resultados obtidos da literatura (Dullaert et al., 2002) em relação à metaheurística *scatter search*.

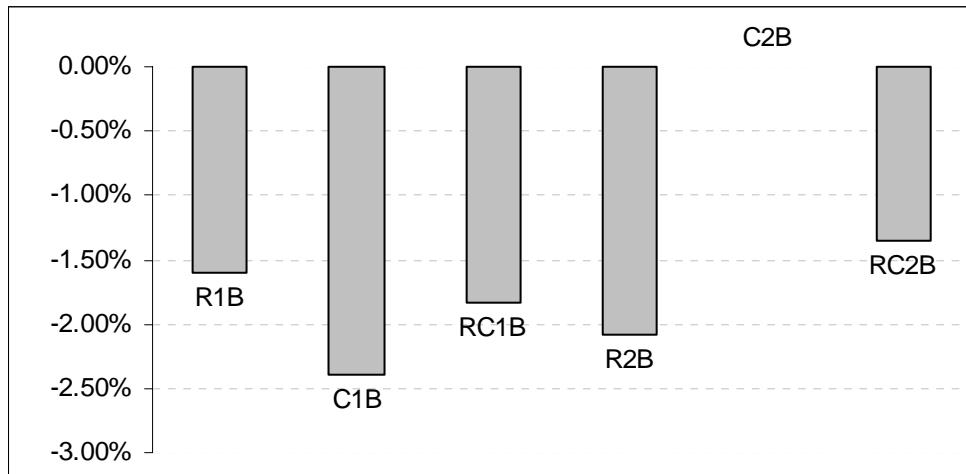
Gráfico 7.6 – Diferença percentual para as classes do conjunto A



De acordo com o gráfico 7.6, para todas as classes de problemas do tipo A, com exceção da classe C2, o algoritmo proposto superou os melhores resultados encontrados na literatura. A diferença média percentual em relação aos resultados de Dullaert et al. (2002), para as classes de problemas do tipo A, foi de -1,20%.

O gráfico 7.7 apresenta a diferença média percentual, para as classes R_{1B} , C_{1B} , R_{C1B} , R_{2B} , C_{2B} e R_{C2B} , dos melhores resultados obtidos da literatura (Dullaert et al., 2002) em relação à metaheurística *scatter search*.

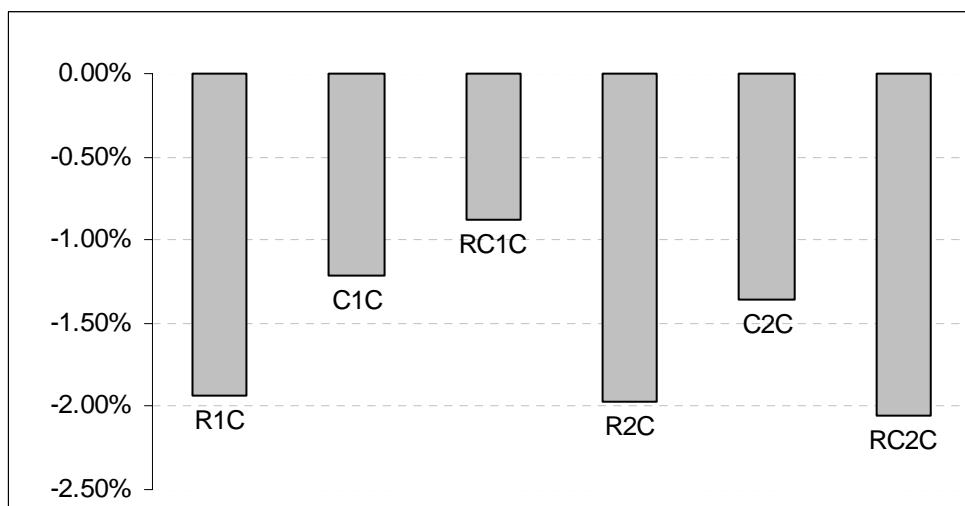
Gráfico 7.7 – Diferença percentual para as classes do conjunto B



De acordo com o gráfico 7.7, o pior resultado obtido foi para a classe C2. As demais classes superaram os melhores resultados obtidos na literatura. A diferença média percentual, da metaheurística *scatter search* em relação aos melhores resultados obtidos na literatura, para o conjunto de problemas do tipo B, foi de $-1,55\%$.

O gráfico 7.8 apresenta a diferença média percentual, para as classes R1_C, C1_C, RC1_C, R2_C, C2_C e RC2_C, dos melhores resultados obtidos da literatura (Dullaert et al., 2002) em relação à metaheurística *scatter search*.

Gráfico 7.8 – Diferença percentual para as classes do conjunto C



De acordo com o gráfico 7.8, podemos concluir que a metaheurística *scatter search* superou os melhores resultados obtidos na literatura para todas as classes de problemas do tipo C. O conjunto de problemas do tipo C apresentou o melhor resultado (diferença média percentual de -1,57%).

A tabela 7.16 apresenta o tempo médio de processamento (segundos) para cada uma das classes de problemas de Liu e Shen (1999).

Tabela 7.16 – Tempo médio de processamento (segundos) para as classes de problemas de Liu e Shen (1999)

R1 _A	R1 _B	R1 _C	C1 _A	C1 _B	C1 _C	RC1 _A	RC1 _B	RC1 _C
530	587	605	644	627	667	617	590	624
R2 _A	R2 _B	R2 _C	C2 _A	C2 _B	C2 _C	RC2 _A	RC2 _B	RC2 _C
672	807	793	676	684	631	834	870	910

CONCLUSÕES E FUTURAS PESQUISAS

O presente trabalho apresentou métodos de resolução para um problema de abastecimento de lojas de um grupo varejista brasileiro. O problema é classificado como um problema de roteirização de veículos com frota heterogênea, janelas de tempo e entregas fracionadas (HFVRPTWSD – *Heterogeneous Fleet Vehicle Routing Problem with Time Windows and Split Deliveries*). O HFVRPTWSD é baseado em um único depósito, a demanda dos clientes pode ser maior que a capacidade dos veículos e, além das restrições de janelas de tempo, há também restrições de capacidade dos veículos e restrições quanto ao tipo de veículo.

Primeiramente, foram propostos quatro algoritmos: heurística de economias, adaptação da heurística de inserção seqüencial de Solomon (1987), heurística de inserção seqüencial de Ho e Haugland (2004) e adaptação da heurística de inserção seqüencial de Dullaert et al. (2002). A partir das heurísticas construtivas, buscou-se implementar a metaheurística *scatter search*. O objetivo, com a implementação das heurísticas e da metaheurística, é minimizar o custo total do frete.

Com relação ao desempenho das heurísticas construtivas, em termos de custo total, o melhor resultado foi obtido pela AHD (adaptação da heurística de Dullaert). Já em relação ao número de veículos, o melhor resultado foi obtido pela AHH (adaptação da heurística de Ho e Haugland). Verificou-se que a utilização de heurísticas construtivas é uma forma simples e rápida de se obter melhorias nas soluções quando comparadas àquelas obtidas empiricamente.

A adoção da metaheurística *scatter search* permitiu a busca de novos espaços de soluções, fazendo com que a solução não fique restrita a ótimos locais. Desta forma, foi possível uma redução ainda maior no custo total da frota, comparado com a solução atual da empresa.

Posteriormente, com o objetivo de avaliar o desempenho dos algoritmos em relação às características dos problemas (frota heterogênea, janelas de tempo e entregas fracionadas), os

mesmos foram adaptados para resolução de três instâncias de problemas da literatura: Solomon (1987), Ho e Haugland (2004) e Liu e Shen (1999).

Para algumas instâncias dos três conjuntos de problemas da literatura, os algoritmos apresentaram resultados similares ou superiores às melhores soluções encontradas na literatura.

Finalmente, é importante indicar que os métodos utilizados neste trabalho podem ser objetos de novas pesquisas, podendo ser adaptados a outros problemas de roteirização de veículos, como problemas com múltiplos depósitos. Os algoritmos também podem ser adaptados a diferentes categorias de produtos. Alguns novos critérios e restrições também podem ser adicionados, como a priorização de pedidos.

Sugere-se, também, como futuras pesquisas, a exploração de novas alternativas de caminhos na malha logística, como o uso do *backhaul*, com o objetivo de minimizar o custo total de frete. O objetivo, com a implementação do *backhaul*, é minimizar as viagens vazias de retorno (não-produtivas) que os veículos fazem no sistema. Assim, buscam-se novos caminhos que atendam mais de um fluxo em cada rota, por exemplo, o veículo sai cheio de um CD (Centro de Distribuição) com destino a uma loja, desta loja percorre vazio até um fornecedor e deste fornecedor volta cheio ao CD.

O algoritmo também pode ser implementado em uma nova instância de problemas da literatura.

REFERÊNCIAS BIBLIOGRÁFICAS

ACHUTHAN, N. R.; CACCETTA, L.; HILL, S.P. An Improved Branch-and-Cut Algorithm for the Capacitated Vehicle Routing Problem. **Transportation Science**, v.37, n.2, p.153-169, 2003.

ALEGRE, J.; LAGUNA, M.; PACHECO, J. **Optimizing the Periodic Pick-up of Raw Materials for a Manufacturer of Auto Parts**, March, 2004.

ALTINKEMER, K.; GAVISH, B. Parallel savings based heuristics for the delivery problem. **Operations Research**, v.39, n.3, p.456-469, 1991.

ARCHETTI, C.; SAVELSBERGH, M.W.P.; SPERANZA, M.G. Worst-Case Analysis of Split Delivery Routing Problems. Department of Quantitative Methods, University of Brescia and School of Industrial and Systems Engineering, Georgia Institute of Technology, 2003a. Aceito em **Transportation Science**.

ARCHETTI, C.; HERTZ, A.; SPERANZA, M.G. **A Tabu Search Algorithm for the Split Delivery Vehicle Routing Problem**. Les Cahiers du GERAD, 2003b. Aceito em **Transportation Science**.

ARCHETTI, C.; MANSINI, M.; SPERANZA, M.G. Complexity and Reducibility of the Skip Delivery Problem. **Transportation Science**, v.39, n.2, p.182-187, 2005.

ASSAD, A. A. Modeling and implementation issues in vehicle routing. In: **Vehicle Routing: Methods and Studies**, GOLDEN, B.L.; ASSAD, A.A. (Eds), Elsevier Sciences Publishers, North-Holland, Amsterdam, p.7-45, 1988.

BADEAU, P.; GUERTIN, F.; GENDREAU, M.; POTVIN, J.Y; TAILLARD, E. A Parallel Tabu Search Heuristic for the Vehicle Routing Problem with Time Windows. **Transportation Research Part C: Emerging Technologies**, v.5, n.2, p.109-122, 1997.

BAKER, E.K. Vehicle routing with time windows constraints. **Logistics and Transportation Review**, v.18, n.4, p.385-401, 1982.

BAKER, B.M.; AYECHEW, M.A. A genetic algorithm for the vehicle routing problem. **Computers & Operations Research**, v.30, n.5, p.787-800, 2003.

BALAKRISHNAN, N. Simple Heuristics for the Vehicle Routing Problem with Soft Time Windows. **Journal of the Operational Research Society**, v.44, n.3, p.279-287, 1993.

BALL, M.; MAGAZINE, M. The design and analysis of heuristics. **Networks**, v.11, n.2, p.215-219, 1981.

BALLOU, R. H. **Logística empresarial:** transporte, administração de materiais e distribuição física. Tradução Hugo T. Y. Y. São Paulo: Atlas, 1993.

BALLOU, R. H. **Gerenciamento da Cadeia de Suprimentos:** Planejamento, Organização e Logística Empresarial. São Paulo: Bookman, 2001.

BARBAROSOGLU, G.; OZGUR, D. A tabu search algorithm for the vehicle routing problem. **Computers & Operations Research**, v.26, p.255-270, n.3, 1999.

BARD, J.F.; KONTORAVDIS, G.; YU, G. A Branch-and-Cut Procedure for the Vehicle Routing Problem with Time Windows. **Transportation Science**, v.36, n.2, p.250-269, 2002.

BELENGUER, J.M.; MARTINEZ, M.C.; MOTA, E. A Lower Bound for the Split Delivery Vehicle Routing Problem. **Operations Research**, v.48, n.5, p.801-810, 2000.

BODIN, L.; BERMAN, L. Routing and scheduling of school buses by computer. **Transportation Science**, v.13, n.2, p.113-129, 1979.

BODIN, L.D.; GOLDEN, B. Classification in Vehicle Routing and Scheduling. **Networks**, v.11, n.2, p.97-108, 1981.

BODIN, L. D.; GOLDEN, B.; ASSAD, A.; BALL, M. Routing and scheduling of vehicle and crews: The state of the art. **Computers & Operations Research**, v.10, n.2, p.63-211, 1983.

BODIN, L.D. Twenty Years of Routing and Scheduling. **Operations Research**, v.38, n.4, p.571-574, 1990.

BOTT, K.; BALLOU, R.H. Research Perspectives in Vehicle Routing and Scheduling. **Transportation Research Part A: Policy and Practice**, v.20A, n.3, p.239-243, 1986.

BRANDÃO, J.; MERCER, A. A tabu search algorithm for the multi-trip vehicle routing and scheduling problem. **European Journal of the Operational Research**, v.100, n.1, p.180-191, 1997.

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

CAMPOS, V.; GLOVER, F.; LAGUNA, M.; MARTÍ, R. An Experimental Evaluation of a Scatter Search for the Linear Ordering Problem. **Journal of Global Optimization**, v.21, n.4, p.397-414, 2001.

CAMPOS, V.; LAGUNA, M.; MARTI, R. Context-independent scatter search and tabu search for permutation problems, 2003. Aceito em **INFORMS Journal on Computing**.

CAMPOS, G. G. **Resolução de um problema de abastecimento com auxílio de metaheurísticas e computação paralela**. São Paulo, 2004. 159 p. Trabalho de conclusão de curso (Departamento de Engenharia de Produção) – Escola Politécnica, Universidade de São Paulo, 2004.

CHIANG, W.C.; RUSSELL, R.A. A Reactive Tabu Search Metaheuristic for the Vehicle Routing Problem with Time Windows. **INFORMS Journal on Computing**, v.9, n.4, p.417-430, 1997.

CHRISTOFIDES, N.; MINGOZZI, A.; TOTH, P. **The Vehicle Routing Problem**. Urbino Working Paper, July, 1978.

CHRISTOFIDES, N.; MINGOZZI, A.; TOTH, P. The vehicle routing problem. In: **Combinatorial Optimization**, CHRISTOFIDES, N.; MINGOZZI, A.; TOTH, P.; SANDI, C. (Eds.), John Wiley, Chichester, New York, p.315-338, 1979.

CHRISTOFIDES, N.; MINGOZZI, A.; TOTH, P. Exact Algorithms for the Vehicle Routing Problem, Based on Spanning Tree and Shortest Path Relaxations. **Mathematical Programming**, v.20, p.255-282, 1981.

CHRISTOFIDES, N. Vehicle Routing. In: **The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization**, LAWER, E.L.; LENSTRA, J.K.; KAN, A.H.G.R.; SHMOYS, D.B (Eds), John Wiley & Sons, 1985.

CHU, F.; LABADI, N.; PRINS, C. A Scatter Search for the periodic capacitated arc routing problem. **European Journal of Operational Research**, v.169, n.2, p.586-605, 2006.

CLARKE, G.; WRIGHT, J.W. Scheduling of Vehicles from a Central Depot to a Number of Delivery Points. **Operations Research**, v.12, n.4, p.568 -581, 1964.

CORBERÁN, A.; FERNÁNDEZ, E.; LAGUNA, M.; MARTÍ, R. Heuristic solutions to the problem of routing school buses with multiple objectives. **Journal of the Operational Research Society**, v.53, n.4, p.427-435, 2002.

CUNG, V-D., MAUTOR, T., MICHELON, P., TAVARES, A Scatter Search Based Approach for the Quadratic Assignment Problem. In: Bäck, T., Michalewicz, Z., Yao, X. (Eds.), Proceedings of IEEE-ICEC-EPS'97, IEEE International Conference on Evolutionary Computation and Evolutionary Programming Conference, p.165–170, 1997.

CUNHA, C. B. **Uma Contribuição para o Problema de Roteirização de Veículos com Restrições Operacionais**. São Paulo, 1997. 222 p. Tese (Doutorado em Engenharia de Transportes) – Escola Politécnica, Universidade de São Paulo, 1997.

CUNHA, C. B. Aspectos práticos da aplicação de modelos de roteirização de veículos a problemas reais. **Transportes**, Rio de Janeiro, v.8, n.2, p.51-74, 2000.

CUNHA, C. B. Um Modelo Matemático para o Problema de Sequenciamento e Programação de Visitas de Gerentes de Banco. **Gestão & Produção**, São Carlos, v.10, n.2, p.183-196, 2003.

DANTZIG, G.B.; FULKERSON, D.R.; JOHNSON, S.M. Solution of a large-scale traveling salesman problem. **Operations Research**, v.2, p.393-410, 1954.

DESROCHERS, M.; VERHOOG, T.W. **A Matching Based Savings Algorithm for the Vehicle Routing Problem**. Les Cahiers du GERAD G-89-04. École des Hautes Études Commerciales de Montréal, 1989.

DESROCHERS, M.; LENSTRA, J.K.; SAVELSBERGH, M.W.P. A classification scheme for the vehicle routing and scheduling problems. **European Journal of Operational Research**, v.46, n.3, p.322-332, 1990.

DESROCHERS, M.; VERHOOG, T.W. A new heuristic for the fleet size and mix vehicle routing problem. **Computers & Operations Research**, v.18, n.3, p.263-274, 1991.

DESROCHERS, M.; DESROSIERS, J.; SOLOMON, M. A new optimization algorithm for the vehicle routing problem with time windows. **Operations Research**, v.40, n.2, p.342-354, 1992.

DROR, M.; TRUDEAU, P. Savings by Split Delivery Routing. **Transportation Science**, v.23, n.2, p.141-145, 1989.

DROR, M.; TRUDEAU, P. Split Delivery Routing. **Naval Research Logistics**, v.37, n.3, p.383-402, 1990.

DROR, M.; LAPORTE, G.; TRUDEAU, P. Vehicle routing with split deliveries. **Discrete Applied Mathematics**, v.50, n.3, p.229-254, 1994.

DULLAERT, W.; JANSENS, G.K.; SÖRENSEN, K.; VERNIMMEN, B. New heuristics for the Fleet Size and Mix Vehicle Routing Problem with Time Windows. **Journal of the Operational Research Society**, v.53, n.11, p.1232-1238, 2002.

FEO, T.A.; RESENDE, M.G.C. A probabilistic heuristic for a computationally difficult set covering problem. **Operations Research Letters**, v.8, n.2, p.67-71, 1989.

FERLAND, J.A.; MICHELON, P. The vehicle scheduling problem with multiple vehicle types. **Journal of the Operational Research Society**, v.39, n.6, p.577-583, 1988.

FISHER, M.; JAIKUMAR, R. A Generalized Assignment Heuristics For Vehicle Routing. **Networks**, v.11, n.2, p.109-124, 1981.

FLEURENT, C.; GLOVER, F.; MICHELON, P.; VALLI, Z. **A Scatter Search Approach for Unconstrained Continuous Optimization**. Proceedings of the 1996 IEEE International Conference on Evolutionary Computation, p.643-648, 1996.

FRIZZELL, P.W.; GIFFIN, J.W. The bounded split delivery vehicle routing problem with grid network distances. **Asia Pacific Journal of Operational Research**, v.9, n.1, p.101-106, 1992.

FRIZZELL, P.W.; GIFFIN, J.W. The Split Delivery Vehicle Scheduling Problem with Time Windows and Grid Network Distances. **Computers & Operations Research**, v.22, n.6, p.655-667, 1995.

GARCIA, B.L.; POTVIN, J.Y.; ROUSSEAU, J.M. A Parallel implementation of the tabu search heuristic for vehicle routing problems with time window constraints. **Computers & Operations Research**, v.21, n.9, p.1025-1033, 1994.

GENDREAU, M.; HERTZ, A.; LAPORTE, G. New Insertion and Postoptimization Procedures for the Traveling Salesman Problem. **Operations Research**, v.40, n.6, p.1086-1094, 1992.

GENDREAU, M.; HERTZ, A.; LAPORTE, G. A tabu search heuristic for the vehicle routing problem. **Management Science**, v.40, n.10, p.1276-1290, 1994.

GENDREAU, M.; LAPORTE, G.; MUSARAGANYI, C.; TAILLARD, E.D. A tabu search heuristic for the heterogeneous fleet vehicle routing problem. **Computers & Operations Research**, v.26, n.12, p.1153-1173, 1999.

GHEYSENS, F.; GOLDEN, B.; ASSAD, A. A comparison of techniques for solving the fleet size and mix vehicle routing problem. **Operations Research Spectrum**, v.6, n.4, p.207-216, 1984.

GHEYSENS, F.; GOLDEN, B.; ASSAD, A. A new heuristic for determining fleet size and composition. **Mathematical Programming Study**, v.26, p.233-236, 1986.

GILLET, B.L.; MILLER, L. A heuristic algorithm for the vehicle dispatch problem. **Operations Research**, v.22, n.4, p.340-349, 1974.

GLOVER, F. Heuristics for Integer Programming Using Surrogate Constraints. **Decision Sciences**, v.8, n.1, p.156-166, 1977.

GLOVER, F. Tabu Search for Nonlinear and Parametric Optimization (with Links to Genetic Algorithms). **Discrete Applied Mathematics**, v.49, n.1-3, p.231-255, 1994.

GLOVER, F.; LAGUNA, M. **Tabu Search**. Kluwer Academic Publishers. Boston, 1997.

GLOVER, F. A Template for Scatter Search and Path Relinking. In: **Artificial Evolution**, Lecture Notes in Computer Science 1363, HAO, J.-K.; LUTTON, E.; RONALD, E.; SCHOENAUER, M.; SNYERS, D. (Eds.), Springer-Verlag, p.13-54, 1998.

GLOVER, F. Scatter Search and Path Relinking. In: **New ideas in Optimization**, CORNE, D.; DORIGO, M.; GLOVER, F. (Eds.), McGraw-Hill, London, p. 297-316, 1999.

GLOVER, F.; LAGUNA, M.; MARTÍ, R. Fundamentals of Scatter Search and Path Relinking. **Control and Cybernetics**, v.29, n.3, p.653-684, 2000.

GLOVER, F.; LAGUNA, M.; MARTÍ, R. Scatter Search. In: **Advances in Evolutionary Computation: Theory and Applications**, GHOSH, A.; TSUTSUI, S. (Eds.), Springer-Verlag, New York, p. 519-537, 2003.

GLOVER, F.; LAGUNA, M.; MARTÍ, R. Scatter Search and Path Relinking: Foundations and Advanced Designs, 2004. In: **New Optimization Techniques in Engineering**, ONWUBOLU, G., BABU, B.V. (Eds.), Springer-Verlag, aceito para publicação.

GOLDBARG, M.C.; LUNA, H.P.L. **Otimização combinatória e programação linear: modelos e algoritmos**. Rio de Janeiro: Campus, 2000.

GOLDEN, B.; BALL, M.; BODIN, L. Current and future research directions in network optimization. **Computers & Operations Research**, v.8, n.2, p. 71-81, 1981.

GOLDEN, B.L.; ASSAD, A.; LEVY, L.; GHEYSENS, F. The fleet size and mix vehicle routing problem. **Computers & Operations Research**, v.11, n.1, p.49-65, 1984.

GOUVÊA, M.T. **Algoritmos para a resolução de um problema geral de roteamento de múltiplos veículos**. São Paulo, 1992. 185 p. Dissertação (Mestrado em Engenharia Naval) – Escola Politécnica, Universidade de São Paulo, 1992.

HO, S.C.; HAUGLAND, D. A tabu search heuristic for the vehicle routing problem with time windows and split deliveries. **Computers & Operations Research**, v.31, n.12, p.1947-1964, 2004.

JAIN, A.S.; MEERAN, S. A multi-level hybrid framework applied to the general flow-shop scheduling problem. **Computers & Operations Research**, v.29, n.13, p.1873-1901, 2002.

KELLY, J.P.; XU, J. A set-partitioning-based heuristic for the vehicle routing problem. **INFORMS Journal of Computing**, v.11, n.2, p.209-224, 1999.

KELLY, J.; RANGASWAMY, B.; XU, J. A Scatter Search-Based Learning Algorithm for Neural Network Training. **Journal of Heuristics**, v.2, n.2, p.129-146, 1996.

KOLEN, A.W.J.; RINNOOY, A.H.G.; TRIENEKENS, H.W.J.M. Vehicle routing with time windows. **Operations Research**, v.35, n.2, p.266-273, 1987.

KONTORAVDIS, G.; BARD, J.F. A GRASP for the vehicle routing problem with time windows. **ORSA Journal on Computing**, v.7, n.1, p.10-23, 1995.

KOSKOSIDIS, Y. A.; POWELL, W.B.; SOLOMON, M.M. An Optimization-Based Heuristic for Vehicle Routing and Scheduling with Soft Time Windows Constraints. **Transportation Science**, v.26, n.2, p.69-85, 1992.

LAGUNA, M. **Scatter Search**. Research Report, University of Colorado, Boulder, 1999.

LAGUNA, M.; MARTÍ, R.; CAMPOS, V. Intensification and Diversification with Elite Tabu Search Solutions for the Linear Ordering Problem. **Computers & Operations Research**, v.26, n.12, p.1217-1230, 1999.

LAGUNA, M.; LOURENÇO, H.; MARTÍ, R. Assigning Proctors to Exams with Scatter Search. In: **Computing Tools for Modeling, Optimization and Simulation: Interfaces in Computer Science and Operations Research**, LAGUNA, M.; GONZÁLEZ-VELARDE, J.L. (Eds.), Kluwer Academic Publishers, p.215-227, 2000.

LAGUNA, M. Scatter Search. In: **Handbook of Applied Optimization**, PARDALOS, P. M.; RESENDE, M. G. C. (Eds.), Oxford University Press, p. 183-193, 2002.

LAGUNA, M.; MARTÍ, R. **Scatter Search: Methodology and implementation** in C. Kluwer Academic Publishers, Boston, 2003.

LAGUNA, M.; MARTÍ, R. Experimental Testing of Advanced Scatter Search Designs for Global Optimization of Multimodal Functions, 2004. Aceito para publicação no **Journal of Global Optimization**.

LANDEGHEM, H.R.G. A bi-criteria heuristic for the vehicle routing problem with time windows. **European Journal of Operational Research**, v.36, n.2, p.217-226, 1988.

LAPORTE, G. The vehicle routing problem: an overview of exact and approximate algorithms. **European Journal of Operational Research**, v.59, n.3, p.345-358, 1992.

LAU, H.C.L.; SIM, M.; TEO, K.M. Vehicle routing problem with time windows and a limited number of vehicles. **European Journal of Operational Research**, v.148, n.3, p.559-569, 2003.

LENSTRA, J.K.; RINNOOY KAN, A.H.G. Complexity of Vehicle and Scheduling Problems. **Networks**, v.11, n.2, p.221-227, 1981.

LIU, F.H.; SHEN, S.Y. The fleet size and mix vehicle routing problem with time windows. **Journal of Operational Research Society**, v.50, n.7, p.721-732, 1999.

MAGNANTI, T. Combinatorial Optimization and Vehicle Fleet Planning: Perspectives and Prospects. **Networks**, v.11, n.2, p.179-214, 1981.

MARTÍ, R.; LOURENÇO, H.; LAGUNA, M. Assigning proctors to exams with scatter search. In: **Computing Tools for Modeling, Optimization and Simulation**, LAGUNA, M.; VELAVERDE, J.L.G. (Eds.), Kluwer Academia Publishers, p.215-227, 2000.

MARTÍ, R.; LAGUNA, M. SCATTER SEARCH: Diseño Básico y Estratégias Avanzadas. **Revista Iberoamericana de Inteligência Artificial**, v.19, n.2, p.123-130, 2003.

MARTÍ, R.; LAGUNA, M.; CAMPOS, V. Scatter search vs genetic algorithms: an experimental evaluation permutation problems, 2002. In: **Adaptive memory and evolution: Tabu search and scatter search**, REGO, C.; ALIDAEE, B. (Eds.), Kluwer Academic Publishers, aceito para publicação.

MARTÍ, R.; LAGUNA, M.; GLOVER, F. Principles of Scatter Search. **European Journal of Operational Research**, v.169, n.2, p.359-372, 2006.

MOLE, R.H.; JAMESON, S.R. A sequencial route-building algorithm employing a generalized savings criterion. **Operational Research**, v.27, n.2, p.503-511, 1976.

MULLASERIL, P.A.; DROR, M.; LEUNG, J. Split-delivery Routing Heuristics in Livestock Feed Distribution. **Journal of the Operational Research Society**, v.48, n.2, p.107-116, 1997.

NICHOLSON, T. *Optimization in industry, optimization techniques*. Longman Group Limited, v.1, 1971.

NOVAES, A.G. **Logística e gerenciamento da cadeia de distribuição**. 2ed. Rio de Janeiro: Campos, 2004.

OSMAN, I.H. Metastrategy simulated annealing and tabu search algorithms for the vehicle routing problem. **Annals of Operations Research**, v.41, n.1-4, p.421-451, 1993.

PAESSENS, H. Savings algorithms for the vehicle routing. **European Journal of Operational Research**, v.34, n.3, p.336-344, 1988.

PESTANA, M. H.; GAGEIRO, J.N. **Análise de dados para ciências sociais: A complementaridade do SPSS**. Edições Silabo, Ltda, 2000.

POTVIN, J.Y.; ROUSSEAU, J.M. A parallel route building algorithm for the vehicle routing and scheduling problem with time windows. **European Journal of Operational Research**, v.66, n.3, p.331-340, 1993.

POTVIN, J.Y.; ROUSSEAU, J.M. An exchange heuristic for routing problems with time windows. **Journal of the Operational Research Society**, v.46, n.12, p.1433-1446, 1995.

POTVIN, J.Y.; BENGIO, S. The Vehicle Routing Problem with Time Windows – Part II: Genetic Search. **INFORMS Journal on Computing**, v.8, n.2, p.165-172, 1996.

POTVIN, J.Y.; KERVAHUT, T.; GARCIA, B.L.; ROUSSEAU, J.M. The Vehicle Routing Problem with Time Windows – Part I: Tabu Search. **INFORMS Journal on Computing**, v.8, n.2, p.158-172, 1996.

PSARAFTIS, H.N. k-Interchange procedures for local search in a precedence-constrained routing problem. **European Journal of Operational Research**, v.13, n.4, p.391-402, 1983.

REEVES, C.R. **Modern Heuristic Techniques for Combinatorial Problems**. John Wiley & Sons. Inc. New York, NY, 1993.

REGO, C.; LEÃO, P. A Scatter Search Tutorial for Graph-Based Permutation Problems. Hearin Center for Enterprise, University of Mississippi, HCES-10-00, USA, 2000. In: **Metaheuristic Optimization via Memory and Evolution: Tabu Search and Scatter Search**, REGO, C; ALIDAEE, B. (Eds.), Kluwer Academic Publishers, p.1-24, 2005.

ROCHAT, Y.; SEMET, F. A Tabu Search Approach for Delivering Pet Food and Flour in Switzerland. **Journal of the Operational Research Society**, v.45, n.11, p.1233-1246, 1994.

ROCHAT, Y.; TAILLARD, E.D. Probabilistic Diversification and Intensification in Local Search for Vehicle Routing. **Journal of Heuristics**, v.1, n.1, p.147-167, 1995.

RONCONI, D.P. **PRO-5826 Heurísticas de melhoria**. São Paulo. 2004.

RONEN, D. Perspectives on practical aspects of truck routing and scheduling. **European Journal of Operational Research**, v.35, n.2, p.137-145, 1988.

RUSSELL, R.A. Hybrid heuristics for the vehicle routing problem with time windows. **Transportation Science**, v.29, n.2, p.156-166, 1995.

RUSSELL, R.A.; CHIANG, W-C. Scatter search for the vehicle routing problem with time windows. **European Journal of Operational Research**, v.169, n.2, p.606-622, 2006.

SALHI, S.; SARI, M.; SAIDI, D.; TOUATI, N. Adaptation of some vehicle fleet mix heuristics. **Omega**, v.20, n.5-6, p.653-660, 1992.

SALHI, S.; RAND, G.K. Incorporating vehicle routing into the vehicle fleet composition problem. **European Journal of Operational Research**, v.66, n.3, p.313-330, 1993.

SEXTON, T.R.; CHOI, Y. Pickup and delivery of partial loads with soft time windows. **American Journal of Mathematical and Management Sciences**, v.6, n.3-4, p.369-398, 1986.

SILVER, E.A.; VIDAL, R.V.V.; WERRA, D. A tutorial on heuristic methods. **European Journal of Operational Research**, v.5, n.3, p.153-162, 1980.

SOLOMON, M. M. On the worst-case performance of some heuristics for the vehicle routing and scheduling with time windows constraints. **Networks**, v.16, n.2, p.161-174, 1986.

SOLOMON, M. M. Algorithms for the Vehicle Routing and Scheduling Problems with Time Windows Constraints. **Operations Research**, v.35, n.2, p.254-265, 1987.

SOLOMON, M.M.; DESROSIERS, J. Time Window Constrained Routing and Scheduling Problem. **Transportation Science**, v.22, n.1, p.1-13, 1988.

SOLOMON, M. M.; BAKER, E.K.; SCHAFER, J.R. Vehicle routing and scheduling problems with time window constraints: Efficient implementations of solution improvement procedures. In: **Vehicle routing: Methods and Studies**, Golden, B.L.; ASSAD, A.A. (Eds.), North-Holland, Amsterdam, p.85-106, 1988.

SOUZA, P.S. **Asynchronous organizations for multi-algorithms problems**. Pittsburgh, 1993. 139 p. Tese (Doutorado em Engenharia) – Carnegie Mellon University, Department of Electrical and Computer Engineering, 1993.

TAILLARD, É.D. Parallel iterative search methods for vehicle routing problems. **Networks**, v.23, n.8, p.661-673, 1993.

TAILLARD, É.D. A diversification/intensification technique for local searches applied to vehicle routing problem. Internal report, 1994.

TAILLARD, É.D.; LAPORTE, G.; GENDREAU, M. Vehicle routeing with multiple use of vehicles. **Journal of the Operational Research Society**, v.47, n.8, p.1065-1070, 1996.

TAILLARD, E.; BADEAU, P.; GENDREAU, M.; GUERTIN, F.; POTVIN, J.Y. A Tabu Search Heuristic for the Vehicle Routing Problem with Soft Time Windows. **Transportation Science**, v.31, n.2, p.170-185, 1997.

TAILLARD, É.D. A heuristic column generation method for the heterogeneous fleet VRP. **RAIRO Recherche Opérationnelle**, v.33, n.1, p.1-14, 1999.

TAN, K. C; LEE, L.H.; ZHU, Q. L.; OU, K. Heuristic Methods for vehicle routing problem with time windows. **Artificial Intelligence in Engineering**, v.15, n.3, p. 281-295, 2001.

TARANTILIS, C.D.; KIRANOUDIS, C.T.; VASSILIADIS, V.S. A threshold accepting metaheuristic for the heterogeneous fixed fleet vehicle routing problem. **European Journal of Operational Research**, v.152, n.1, p.148-158, 2004.

TEIXEIRA, R. G.; CUNHA, C. B. Heurísticas para o problema de dimensionamento e roteirização de uma frota heterogênea utilizando o algoritmo out-of-kilter. **Transportes**, v.10, n.2, 2002.

THOMPSON, P.M.; PSARAFTIS, H.N. Cyclic Transfer Algorithms for Multivehicle Routing and Scheduling Problems. **Operations Research**, v.41, n.5, 1993.

XU, J., CHIU, S., GLOVER, F. Tabu Search and Evolutionary Scatter Search for ‘Tree-Star’ Network Problems, with Applications to Leased-Line Network Design. In: **Telecommunications Optimization: Heuristic and Adaptive Techniques**, CORNE, D.W; OATES, M. J.; SMITH, G. D. (Eds.), Wiley, 2000.

YAMASHITA, D. S. **Scatter Search para Programação de Projetos com Custo de Disponibilidade de Recursos sob Incerteza**. Campinas, 2003. Tese (Doutorado em Engenharia de Sistemas) – Faculdade de Engenharia Elétrica e Computação, Unicamp, 2003.

YAMASHITA, D.S.; ARMENTANO, V.A.; LAGUNA, M. Scatter Search for Project scheduling with resource availability cost. **European Journal of Operational Research**, v.169, n.2, p.623-637, 2006.

WASSAN, N.A.; OSMAN, I.H. Tabu search variants for the mix fleet vehicle routing problem. **Journal of the Operational Research Society**, v.53, n.7, p.768-782, 2002.

ZANAKIS, S.H.; EVANS, J.R.; VAZACOPOULOS, A.A. Heuristic methods and applications: a categorized survey. **European Journal of Operational Research**, v.43, n.1, p.88-110, 1989.

APÊNDICE A – Resultados detalhados

Rota	Roteiro	Veiculo	Distancia	Tempo	Custo
1	0-191-0	Truck	434,85	7,5	416,55
2	0-531-0	Truck	66,6	2,2	163,38
3	0-606-0	Carreta	3651,65	63,0	3805,06
4	0-963-0	Carreta	976,76	16,8	1139,54
5	0-963-0	Carreta	976,76	16,8	1139,54
6	0-963-0	Carreta	976,76	16,8	1139,54
7	0-963-0	Carreta	976,76	16,8	1139,54
8	0-17-0	Truck	47,57	2,4	150,29
9	0-224-0	Truck	36,35	1,8	142,58
10	0-278-0	Truck	474,81	8,2	444,02
11	0-280-0	Truck	204,18	3,5	257,97
12	0-282-0	Truck	204,18	3,5	257,97
13	0-302-0	Truck	38,83	1,9	144,29
14	0-302-0	Truck	38,83	1,9	144,29
15	0-1304-0	Carreta	1224,93	21,1	1386,84
16	0-307-0	Carreta	19,36	1,0	185,49
17	0-309-0	Carreta	59,68	2,0	225,67
18	0-309-0	Carreta	59,68	2,0	225,67
19	0-310-0	Carreta	50,15	1,7	216,17
20	0-310-0	Carreta	50,15	1,7	216,17
21	0-311-0	Carreta	178,75	4,0	344,32
22	0-314-0	Carreta	73,06	2,4	239
23	0-314-0	Carreta	73,06	2,4	239
24	0-315-0	Carreta	61,63	2,1	227,61
25	0-315-0	Carreta	61,63	2,1	227,61
26	0-318-0	Carreta	841,03	14,5	1004,29
27	0-319-0	Carreta	63,22	2,1	229,2
28	0-319-0	Carreta	63,22	2,1	229,2
29	0-320-0	Truck	75	2,5	169,16
30	0-321-0	Truck	82,63	2,8	174,4
31	0-322-0	Truck	60,87	2,0	159,44
32	0-322-0	Truck	60,87	2,0	159,44
33	0-323-0	Truck	72,45	2,4	167,4
34	0-323-0	Truck	72,45	2,4	167,4
35	0-328-0	Carreta	55,57	1,9	221,57
36	0-329-0	Carreta	42,96	2,1	209,01
37	0-330-0	Carreta	1224,93	21,1	1386,84
38	0-337-0	Truck	51,44	1,7	152,96
39	0-337-0	Truck	51,44	1,7	152,96
40	0-337-0	Truck	51,44	1,7	152,96
41	0-337-0	Truck	51,44	1,7	152,96
42	0-338-0	Truck	77,31	2,6	170,74
43	0-339-0	Carreta	5873,3	101,3	6021,93
44	0-341-0	Carreta	38,97	1,9	205,04

Rota	Roteiro	Veiculo	Distancia	Tempo	Custo
45	0-341-0	Carreta	38,97	1,9	205,04
46	0-344-0	Carreta	5873,3	101,3	6021,93
47	0-346-0	Carreta	171,51	3,8	337,11
48	0-347-0	Carreta	2146,21	37,0	2304,9
49	0-351-0	Carreta	36,42	1,8	202,49
50	0-353-0	Carreta	3651,65	63,0	3805,06
51	0-355-0	Carreta	2146,21	37,0	2304,9
52	0-357-0	Carreta	25,48	1,3	191,59
53	0-359-0	Carreta	48,7	2,4	214,73
54	0-612-0	Truck	57,95	1,9	157,43
55	0-612-0	Truck	57,95	1,9	157,43
56	0-717-0	Truck	48,46	2,4	150,91
57	0-746-0	Truck	150,56	3,3	221,1
58	0-785-0	Truck	93,71	3,1	182,02
59	0-854-0	Truck	39,25	2,0	144,57
60	0-318-0	Truck	29,68	1,5	138
61	0-333-0	Truck	340,12	5,9	351,43
62	0-333-0	Truck	340,12	5,9	351,43
63	0-336-0	Truck	56,99	1,9	156,77
64	0-336-0	Truck	56,99	1,9	156,77
65	0-336-0	Truck	56,99	1,9	156,77
66	0-349-0	Truck	31,61	1,6	139,33
67	0-355-230-602-0	Carreta	2146,21	37,0	2304,9
68	0-347-0	Carreta	2146,21	37,0	2304,9
69	0-603-0	Carreta	2146,21	37,0	2304,9
70	0-609-0	Truck	823,84	14,2	683,98
71	0-21-0	Toco	682,61	11,8	612,01
72	0-35-0	Truck	682,61	11,8	586,88
73	0-255-339-0	Truck	5876,3	101,3	4157,55
74	0-344-0	Carreta	5873,3	101,3	6021,93
75	0-606-610-0	Carreta	3651,65	63,0	3805,06
76	0-353-0	Truck	3651,65	63,0	2628,1
77	0-330-304-0	Carreta	1224,93	21,1	1386,84
78	0-833-832-831-0	Truck	1222,76	21,1	958,24
79	0-122-0	Truck	1172,95	20,2	924
80	0-402-0	Toco	995	17,2	855,05
81	0-312-349-0	Carreta	596,11	10,3	760,22
82	0-473-0	Truck	331,24	5,7	345,32
83	0-371-329-0	Truck	331,24	5,7	345,32
84	0-838-963-0	Carreta	976,76	16,8	1139,54
85	0-842-841-628-145-635-640-645-0	Carreta	985,54	17,0	1148,29
86	0-681-625-823-822-654-680-0	Truck	1018,95	17,6	818,12
87	0-844-333-143-603-843-835-0	Truck	928,92	16,0	756,23
88	0-361-683-643-332-334-240-241-671-633-0	Carreta	965,25	16,6	1128,07
89	0-335-646-236-650-136-146-0	Carreta	927,72	16,0	1090,67

Rota	Roteiro	Veiculo	Distância	Tempo	Custo
90	0-679-0	Carreta	885,55	15,3	1048,65
91	0-346-474-0	Truck	902,07	15,6	737,77
92	0-325-0	Carreta	138,83	3,1	304,55
93	0-619-630-636-331-638-641-0	Carreta	900,04	15,5	1063,09
94	0-278-276-0	Toco	476,91	8,2	451,98
95	0-370-303-262-318-261-297-0	Carreta	848,27	14,6	1011,5
96	0-260-269-259-369-264-0	Truck	843,7	14,5	697,64
97	0-268-267-0	Truck	838,6	14,5	694,13
98	0-748-217-0	Truck	249,34	4,3	289,02
99	0-776-0	Leve	160,73	3,6	166,09
100	0-327-0	Truck	160,73	3,6	228,1
101	0-191-0	Leve	434,85	7,5	349,75
102	0-277-0	Truck	434,85	7,5	416,55
103	0-270-0	Truck	372,5	6,4	373,68
104	0-273-0	Truck	381,41	6,6	379,81
105	0-28-0	Leve	340,12	5,9	286,29
106	0-333-0	Leve	340,12	5,9	286,29
107	0-274-186-0	Truck	340,12	5,9	351,43
108	0-275-0	Leve	288,02	5,0	251,38
109	0-341-0	Truck	239,47	4,1	282,23
110	0-282-281-0	Leve	204,18	3,5	195,2
111	0-280-279-0	Truck	204,18	3,5	257,97
112	0-342-0	Carreta	149,96	3,3	315,64
113	0-102-52-0	Truck	188,23	4,2	247
114	0-821-756-0	Truck	188,23	4,2	247
115	0-749-0	Truck	188,23	4,2	247
116	0-755-0	Leve	188,23	4,2	184,52
117	0-853-0	Truck	188,23	4,2	247
118	0-293-465-0	Truck	178,75	4,0	240,48
119	0-37-0	Truck	178,75	4,0	240,48
120	0-305-0	Carreta	178,75	4,0	344,32
121	0-324-311-0	Carreta	188,84	4,2	354,38
122	0-602-0	Truck	86,57	2,9	177,11
123	0-692-0	Toco	162,32	3,6	207,22
124	0-348-0	Carreta	162,32	3,6	327,95
125	0-44-0	Leve	162,32	3,6	167,15
126	0-742-0	Truck	162,32	3,6	229,18
127	0-757-746-0	Truck	150,56	3,3	221,1
128	0-777-0	Leve	150,56	3,3	159,28
129	0-741-0	Truck	140,9	3,1	214,46
130	0-740-0	Leve	140,9	3,1	152,81
131	0-759-0	Toco	107,84	2,4	164,84

Rota	Roteiro	Veiculo	Distância	Tempo	Custo
132	0-782-343-0	Truck	149,96	3,3	220,69
133	0-417-72-0	Truck	149,96	3,3	220,69
134	0-781-780-0	Truck	149,96	3,3	220,69
135	0-785-784-0	Leve	93,71	3,1	121,19
136	0-350-0	Carreta	90,27	3,0	256,16
137	0-809-0	Toco	84,97	2,8	147,05
138	0-360-0	Carreta	77,29	2,6	243,22
139	0-2-0	Leve	138,83	3,1	151,42
140	0-734-0	Truck	111,66	2,5	194,36
141	0-352-0	Carreta	97,03	3,2	262,89
142	0-801-739-0	Truck	109,18	2,4	192,65
143	0-462-725-0	Truck	95,2	3,2	183,05
144	0-320-445-0	Toco	81,27	2,7	144,17
145	0-792-0	Toco	93	3,1	153,29
146	0-718-0	Truck	84,98	2,8	176,02
147	0-326-0	Truck	83,12	2,8	174,74
148	0-321-0	Truck	82,63	2,8	174,4
149	0-817-325-0	Truck	82,06	2,7	174,01
150	0-795-796-0	Truck	80,45	2,7	172,9
151	0-793-849-0	Truck	82,01	2,7	173,97
152	0-788-794-0	Truck	78,92	2,6	171,85
153	0-789-0	Leve	67,89	2,3	103,89
154	0-468-338-301-0	Truck	79	2,6	171,9
155	0-736-723-0	Truck	78,74	2,6	171,72
156	0-738-444-0	Truck	66,52	2,2	163,32
157	0-808-0	Truck	76,01	2,5	169,85
158	0-314-0	Carreta	73,06	2,4	239
159	0-712-531-0	Truck	72,35	2,4	167,33
160	0-763-323-0	Truck	85,32	2,8	176,25
161	0-713-0	Toco	69,33	2,3	134,88
162	0-319-0	Toco	63,22	2,1	130,13
163	0-315-0	Leve	61,63	2,1	99,69
164	0-429-428-0	Toco	60,3	2,0	127,86
165	0-516-786-0	Truck	61,98	2,1	160,2
166	0-322-0	Toco	60,87	2,0	128,3
167	0-478-711-0	Truck	60,16	2,0	158,96
168	0-214-0	Leve	54,08	1,8	94,64
169	0-409-0	Truck	53,02	1,8	154,05
170	0-440-705-0	Truck	51,67	1,7	153,11
171	0-201-0	Truck	47,35	2,4	150,14
172	0-70-0	Leve	61,16	2,0	99,38
173	0-309-0	Truck	59,68	2,0	158,62

Rota	Roteiro	Veiculo	Distância	Tempo	Custo
174	0-76-0	Truck	59,07	2,0	158,21
175	0-328-0	Carreta	55,57	1,9	221,57
176	0-851-612-0	Truck	58,45	1,9	157,78
177	0-336-0	Toco	56,99	1,9	125,28
178	0-710-848-0	Truck	57,62	1,9	157,2
179	0-436-289-0	Truck	55,41	1,8	155,69
180	0-107-0	Leve	51,74	1,7	93,07
181	0-715-0	Toco	54,47	1,8	123,32
182	0-790-0	Leve	53,77	1,8	94,43
183	0-308-0	Carreta	53,22	1,8	219,24
184	0-227-762-0	Toco	51,56	1,7	121,06
185	0-337-0	Truck	51,44	1,7	152,96
186	0-461-766-0	Toco	50,89	1,7	120,54
187	0-310-0	Truck	50,15	1,7	152,07
188	0-13-0	Truck	49,5	2,5	151,62
189	0-768-0	Toco	48,98	2,4	119,05
190	0-208-0	Leve	46,21	2,3	89,37
191	0-359-0	Truck	48,7	2,4	151,07
192	0-737-717-0	Truck	49,65	2,5	151,73
193	0-435-703-0	Truck	48,52	2,4	150,95
194	0-476-466-0	Truck	48,17	2,4	150,71
195	0-706-0	Toco	47,86	2,4	118,18
196	0-17-707-0	Toco	47,57	2,4	117,95
197	0-26-0	Toco	45,91	2,3	116,66
198	0-205-0	Leve	44,13	2,2	87,97
199	0-221-11-0	Truck	44,15	2,2	147,95
200	0-410-0	Truck	45,67	2,3	148,99
201	0-854-0	Leve	39,25	2,0	84,7
202	0-170-61-0	Truck	44,82	2,2	148,41
203	0-87-291-0	Truck	45,19	2,3	148,66
204	0-114-0	Toco	43,42	2,2	114,72
205	0-231-0	Leve	40,74	2,0	85,7
206	0-329-0	Carreta	42,96	2,1	209,01
207	0-459-770-0	Truck	43,72	2,2	147,65
208	0-704-0	Toco	40,45	2,0	112,41
209	0-341-0	Carreta	38,97	1,9	205,04
210	0-203-0	Leve	38,31	1,9	84,07
211	0-56-0	Leve	41,76	2,1	86,38
212	0-21-0	Truck	41,28	2,1	145,98
213	0-60-0	Truck	40,48	2,0	145,42
214	0-25-147-0	Truck	40,51	2,0	145,45
215	0-302-0	Truck	38,83	1,9	144,29

Rota	Roteiro	Veiculo	Distância	Tempo	Custo
216	0-1-0	Truck	38,83	1,9	144,29
217	0-204-0	Toco	38,48	1,9	110,88
218	0-779-216-0	Truck	38,74	1,9	144,23
219	0-292-0	Truck	35,85	1,8	142,24
220	0-850-416-0	Truck	38,25	1,9	143,89
221	0-16-0	Toco	37,81	1,9	110,36
222	0-406-226-0	Truck	38,87	1,9	144,31
223	0-464-116-0	Truck	37,29	1,9	143,23
224	0-351-0	Carreta	36,42	1,8	202,49
225	0-183-405-349-224-0	Truck	40,16	2,0	145,2
226	0-20-0	Toco	36,25	1,8	109,15
227	0-50-0	Truck	33,46	1,7	140,6
228	0-446-0	Leve	35,85	1,8	82,42
229	0-412-0	Carreta	30,03	1,5	196,12
230	0-442-729-0	Truck	37,67	1,9	143,49
231	0-469-185-0	Truck	33,68	1,7	140,75
232	0-149-0	Truck	33,57	1,7	140,67
233	0-9-0	Leve	33,65	1,7	80,95
234	0-597-290-0	Truck	32,93	1,6	140,23
235	0-611-0	Leve	16,38	0,8	69,38
236	0-39-0	Leve	31,69	1,6	79,64
237	0-7-0	Leve	31,45	1,6	79,47
238	0-150-0	Truck	30,65	1,5	138,67
239	0-223-0	Leve	28,43	1,4	77,45
240	0-106-62-0	Toco	30,19	1,5	104,43
241	0-6-0	Truck	26,21	1,3	135,61
242	0-318-0	Toco	29,68	1,5	104,03
243	0-356-0	Carreta	23,64	1,2	189,76
244	0-439-48-427-0	Truck	28,59	1,4	137,25
245	0-307-0	Toco	19,36	1,0	96
246	0-317-0	Carreta	26,57	1,3	192,68
247	0-207-0	Truck	25,81	1,3	135,34
248	0-408-0	Toco	25,81	1,3	101,02
249	0-357-0	Carreta	25,48	1,3	191,59
250	0-105-778-0	Truck	19,33	1,0	130,88
251	0-352-0	Toco	21,81	1,1	97,91
252	0-811-0	Leve	17,58	0,9	70,18
253	0-222-0	Truck	21,4	1,1	132,3
254	0-422-0	Truck	21,4	1,1	132,3
255	0-810-858-0	Truck	20,64	1,0	131,78
256	0-88-857-0	Toco	14,84	0,7	92,48
257	0-326-0	Truck	14,58	0,7	127,62
258	0-859-0	Truck	8,18	0,4	123,22

ANEXO A – Exemplo de aplicação de Busca Dipersa

O exemplo do método *scatter search* foi extraído do trabalho de Laguna (1999). Considere o seguinte problema de programação inteira (problema da mochila) com restrições de pesos:

$$\begin{aligned} \text{max } & 11x_1 + 10x_2 + 9x_3 + 12x_4 + 10x_5 + 6x_6 + 7x_7 + 5x_8 + 3x_9 + 8x_{10} \\ \text{s.t. } & 33x_1 + 27x_2 + 16x_3 + 14x_4 + 29x_5 + 30x_6 + 31x_7 + 33x_8 + 14x_9 + 18x_{10} \leq 100 \\ & x_i = \{0,1\} \text{ para } i = 1, \dots, 10. \end{aligned}$$

1. Método de geração de soluções diversas

O método apresentado foi descrito por Glover (1998). Escolhe-se um valor para o parâmetro $h \leq n - 1$, onde n é o número de variáveis do problema. Vamos considerar $h = 5$. Foi escolhido como solução inicial $x = (0, 0, \dots, 0)$. Para cada valor de h , duas soluções são geradas. Soluções do tipo 1 são denominadas x' e iniciam com $x'_i = 0$ para todo i . Assim, alguns elementos são modificados:

$$\begin{aligned} x'_1 &= 1 - x_1 \\ x'_{1+hk} &= 1 - x_{1+hk} \text{ para } k = 1, \dots, n/h. \end{aligned} \tag{1}$$

Soluções do tipo 2 são denominadas x'' e obtidas como um complemento das soluções do tipo 1, $x''_i = 1 - x'_i$. A tabela 2 mostra as 10 soluções geradas de acordo com este método.

Tabela 2 - Soluções diversas geradas em *scatter search*

h	x'	x''
1	(1,1,1,1,1,1,1,1,1,1)	(0,0,0,0,0,0,0,0,0,0)
2	(1,0,1,0,1,0,1,0,1,0)	(0,1,0,1,0,1,0,1,0,1)
3	(1,0,0,1,0,0,1,0,0,1)	(0,1,1,0,1,1,0,1,1,0)
4	(1,0,0,0,1,0,0,0,1,0)	(0,1,1,1,0,1,1,1,0,1)
5	(1,0,0,0,0,1,0,0,0,0)	(0,1,1,1,1,0,1,1,1,1)

2. Método de melhoria

O método de geração das soluções diversas não garante soluções factíveis. A partir das 10 soluções obtidas da tabela 2 aplica-se o método de melhoria. São consideradas duas operações:

- a) Se a solução inicial é infactível, o método primeiramente torna a solução factível. Isto é feito trocando valores de um para zero até que a restrição seja satisfeita. Inicia-se com a variável que tem a menor taxa lucro-por-peso. Uma vez que a solução se torna factível o passo seguinte é aplicado.
- b) Se a solução inicial é factível, busca-se melhoria trocando valores de zero para um. A primeira variável considerada é a que tem a maior taxa lucro-por-peso. O processo termina quando não há nenhuma variável que pode mudar o valor de 0 para 1 sem violar a restrição.

A tabela 3 mostra as soluções obtidas após a aplicação do método de melhoria.

Tabela 3 - Soluções iniciais e soluções após melhoria

Soluções	Solução inicial	Função Objetivo	Solução melhorada	Função Objetivo
1	(1,1,1,1,1,1,1,1,1)	81+	(0,1,1,1,0,0,0,0,1,1)	42
2	(1,0,1,0,1,0,1,0,1,0)	40+	(1,0,1,1,1,0,0,0,0,0)	42
3	(1,0,0,1,0,0,1,0,0,1)	38	(1,0,0,1,0,0,1,0,0,1)	38
4	(1,0,0,0,1,0,0,0,1,0)	24	(1,0,0,1,1,0,0,0,1,0)	36
5	(1,0,0,0,0,1,0,0,0,0)	17	(1,0,1,1,0,1,0,0,0,0)	38
6	(0,0,0,0,0,0,0,0,0,0)	0	(0,1,1,1,0,0,0,0,0,1)	39
7	(0,1,0,1,0,1,0,1,0,1)	41+	(0,1,0,1,0,1,0,0,0,1)	36
8	(0,1,1,0,1,1,0,1,1,0)	43+	(0,1,1,1,1,0,0,0,1,0)	44*
9	(0,1,1,1,0,1,1,1,0,1)	57+	(0,1,1,1,0,0,0,0,1,1)	42
10	(0,1,1,1,1,0,1,1,1,1)	64+	(0,1,1,1,0,0,0,0,1,1)	42

Legenda:

+ solução infactível

* solução incumbente

Note que as soluções 1, 9 e 10 na tabela 3 apresentaram a mesma solução após melhoria, ambas com valor 42 na função objetivo. A tabela 4 mostra a aplicação do método de melhoria para a solução ótima (solução 8).

Tabela 4 - Iterações do método de melhoria

Iteração	Solução corrente	Função objetivo	Peso total	Candidatas (lucro/peso)	Movimento selecionado
1	(0,1,1,0,1,1,0,1,1,0)	43	149	x2 = 0 (0,370) x3 = 0 (0,563) x5 = 0 (0,345) x6 = 0 (0,200) x8 = 0 (0,152) x9 = 0 (0,214)	x8 = 0
2	(0,1,1,0,1,1,0,0,1,0)	38	116	x2 = 0 (0,370) x3 = 0 (0,563) x5 = 0 (0,345) x6 = 0 (0,200) x9 = 0 (0,214)	x6 = 0
3	(0,1,1,0,1,0,0,0,1,0)	32	86	x4 = 1 (0,857)	x4 = 1
4	(0,1,1,1,1,0,0,0,1,0)	44	100	Nenhuma	

3. Método de criação e atualização do conjunto de referência

O conjunto de referência de b soluções é formado por b_1 soluções de alta qualidade e b_2 soluções diversas. Considera-se $b = 5$, onde $b_1 = 3$ e $b_2 = 2$. Desta forma as soluções de alta qualidade são as soluções 1, 2 e 8.

O próximo passo é determinar as soluções diversas. A distância entre duas soluções é determinada pela soma das diferenças absolutas entre seus valores correspondentes. Por exemplo, a distância entre a solução 1 e 8 após a melhoria é calculada da seguinte forma:

$$\begin{aligned}
 & (0, 1, 1, 1, 0, 0, 0, 0, 1, 1) \\
 & \underline{(0, 1, 1, 1, 1, 0, 0, 0, 1, 0)} \\
 & (0+0+0+0+1+0+0+0+0+1) = 2
 \end{aligned}$$

A medida de distância é utilizada para selecionar as duas soluções mais distantes que farão parte do conjunto das soluções diversas. A solução mais distante é aquela que não pertence ao conjunto de referência e maximiza a distância mínima em relação às soluções do

conjunto de referência. A tabela 5 mostra as distâncias de cada solução candidata para cada solução corrente do conjunto de referência.

Tabela 5 - Distâncias entre soluções

Candidata	Solução	Distância para solução			Distância mínima
		Solução 1	Solução 2	Solução 8	
3	(1,0,0,1,0,0,1,0,0,1)	5	4	7	4
4	(1,0,0,1,1,0,0,0,1,0)	5	3	3	2
5	(1,0,1,1,0,1,0,0,0,0)	5	3	5	2
6	(0,1,1,1,0,0,0,0,0,1)	1	4	3	1
7	(0,1,0,1,0,1,0,0,0,1)	3	6	5	3

A lista de soluções candidatas na tabela 5 não incluem nem a solução 9 nem a solução 10 porque estas soluções são idênticas à solução 1 que já pertence ao conjunto de referência. A primeira solução que maximiza a distância mínima é a solução 3 que é adicionada ao conjunto de referência. A próxima e última solução a ser incluída no *RefSet* é a solução 7 que maximiza a distância mínima das soluções candidatas. Portanto, o conjunto de referência é formado por soluções de alta qualidade (1, 2 e 8) e soluções diversas (3 e 7).

4. Método de geração de subconjuntos

O método pode gerar os seguintes subconjuntos:

- 1) Todos os subconjuntos de 2 soluções.
- 2) Subconjuntos de 3 soluções derivados dos subconjuntos de 2 elementos, incluindo a melhor solução (que não está incluída) em cada subconjunto de 2 elementos.
- 3) Subconjuntos de 4 soluções derivados dos subconjuntos de 3 elementos, incluindo a melhor solução (que não está incluída) em cada subconjunto de 2 elementos.
- 4) Os subconjuntos consistem das i melhores soluções onde $i = 5$ a b .

Os subconjuntos gerados através do conjunto de referência estão listados a seguir.

Tabela 6 - Geração de subconjuntos

Tipo	Subconjuntos
1	(1,2) (1,3) (1,7) (1,8) (2,3) (2,7) (2,8) (3,7) (3,8) (7,8)
2	(1,2,8) (1,3,8) (1,7,8) (2,3,8) (2,7,8) (3,7,8)
3	(1,2,3,8) (1,2,7,8) (1,3,7,8)
4	(1,2,3,7,8)

5. Método de combinação de soluções

As soluções são combinadas com base no primeiro método apresentado no item 3.1:

$$Score(k) = \frac{\sum_{j \in S} FO_j \cdot x_k^j}{\sum_{j \in S} FO_j}, \quad k = 1, \dots, m \quad (2)$$

e,

$$\dot{x}_k = \begin{cases} 1 & \text{se } score(k) > 0,5 \\ 0 & \text{se } score(k) \leq 0,5 \end{cases} \quad (3)$$

Para ilustrar o método de combinação de soluções, será considerado o subconjunto do tipo 2 dado pelas soluções 3, 7 e 8. O valor da função objetivo da solução 3 é 38, o valor da função objetivo da solução 7 é 36 e o valor da função objetivo da solução 8 é 44. Assim, $FO(3) = 38$, $FO(7) = 36$, e $FO(8) = 44$. O *score* de cada posição k pode ser calculado, como mostra a tabela 7.

Tabela 7 - Cálculo do *score* e combinação do subconjunto (3,7,8)

Sol.	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
3	0.322	0.000	0.000	0.322	0.000	0.000	0.322	0.000	0.000	0.322
7	0.000	0.305	0.000	0.305	0.000	0.305	0.000	0.000	0.000	0.305
8	0.000	0.373	0.373	0.373	0.373	0.000	0.000	0.000	0.373	0.000
Total	0.322	0.678	0.373	1.000	0.373	0.305	0.322	0.000	0.373	0.627

Após a combinação, a nova solução passa a ser $(0, 1, 0, 1, 0, 0, 0, 0, 0, 1)$ com um valor na função objetivo de 30 e um peso total de 59. Esta solução factível pode ser melhorada através do método de melhoria. Desta forma, o valor de x_5 (que tem a maior taxa lucro-por-peso) muda de zero para um, sem violar a restrição de capacidade, resultando em um valor da função objetivo de 40 e um peso total de 88. Dado que mais nenhuma variável pode mudar de zero para 1 sem violar a restrição de capacidade, então o processo de melhoria é finalizado.

Neste exemplo, uma nova solução combinada passa a fazer parte do conjunto de referência se seu valor da função objetivo é melhor que qualquer uma das soluções de alta qualidade de *RefSet*. Portanto, esta solução passa a compor as soluções do conjunto de referência de alta qualidade. Alternativamente, se a nova solução aumenta a diversidade de *RefSet* ela passa a fazer parte do conjunto de referência de soluções diversas. Se o conjunto de referência é modificado, o método de geração de subconjuntos, o método de combinações e o método de melhoria são aplicados na seqüência. A aplicação do método continua enquanto houver um elemento capaz de entrar no conjunto de referência. Ao final de uma iteração, o método de diversificação pode ser aplicado novamente e a busca continua até atingir o critério de parada.

ANEXO B – Cálculo da distância geodésica

Segundo Ballou (2001), a distância geodésica entre dois pontos A e B é calculada da seguinte forma:

$$D_{AB} = 6340 \left\{ \arccos \left[\sin(\text{Lat}_A) * \sin(\text{Lat}_B) + \cos(\text{Lat}_A) * \cos(\text{Lat}_B) * \cos(\text{Long}_B - \text{Long}_A) \right] \right\}$$

onde: (4)

D_{AB} = distância entre os pontos A e B (km)

Lat_A = latitude do ponto A (radianos)

Long_A = longitude do ponto A (radianos)

Lat_B = latitude do ponto B (radianos)

Long_B = longitude do ponto B (radianos)

ANEXO C – Instâncias de problemas de Liu e Shen (1999)
Tabela 8 – Conjunto de problemas R1 de Liu e Shen (1999)

R1 (36 problemas)				
Veículo	Capacidade	Custo		
		R1a	R1b	R1c
A	30	50	10	5
B	50	80	16	8
C	80	140	28	14
D	120	250	50	25
E	200	500	100	50

Tabela 9 – Conjunto de problemas C1 de Liu e Shen (1999)

C1 (27 problemas)				
Veículo	Capacidade	Custo		
		C1a	C1b	C1c
A	100	300	60	30
B	200	800	160	80
C	300	1350	270	135

Tabela 10 – Conjunto de problemas RC1 de Liu e Shen (1999)

RC1 (24 problemas)				
Veículo	Capacidade	Custo		
		RC1a	RC1b	RC1c
A	40	60	12	6
B	80	150	30	15
C	150	300	60	30
D	200	450	90	45

Tabela 11 – Conjunto de problemas R2 de Liu e Shen (1999)

R2 (33 problemas)				
Veículo	Capacidade	Custo		
		R2a	R2b	R2c
A	300	450	90	45
B	400	700	140	70
C	600	1200	240	120
D	1000	2500	500	250

Tabela 12 – Conjunto de problemas C2 de Liu e Shen (1999)

C2 (24 problemas)				
Veículo	Capacidade	Custo		
		C2a	C2b	C2c
A	400	1000	200	100
B	500	1400	280	140
C	600	2000	400	200
D	700	2700	540	270

Tabela 13 – Conjunto de problemas RC2 de Liu e Shen (1999)

RC2 (24 problemas)				
Veículo	Capacidade	Custo		
		RC2a	RC2b	RC2c
A	100	150	30	15
B	200	350	70	35
C	300	550	110	55
D	400	800	160	80
E	500	1100	220	110
F	1000	2500	500	250