

Universidade Federal do Rio de Janeiro

APLICAÇÃO DE METAHEURÍSTICA HÍBRIDA NA RESOLUÇÃO DO
PROBLEMA DIAL-A-RIDE

Maxwell Scardini Kaiser

2009



COPPE/UFRJ

APLICAÇÃO DE METAHEURÍSTICA HÍBRIDA NA RESOLUÇÃO DO
PROBLEMA DIAL-A-RIDE

Maxwell Scardini Kaiser

Dissertação de Mestrado apresentada ao Programa de Pós-graduação em Engenharia de Transportes, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Mestre em Engenharia de Transportes.

Orientador: Hostilio Xavier Rattton Neto

Rio de Janeiro
Novembro de 2009

APLICAÇÃO DE METAHEURÍSTICA HÍBRIDA NA RESOLUÇÃO DO
PROBLEMA DIAL-A-RIDE

Maxwell Scardini Kaiser

DISSERTAÇÃO SUBMETIDA AO CORPO DOCENTE DO INSTITUTO ALBERTO
LUIZ COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE ENGENHARIA
(COPPE) DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE
DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE
EM CIÊNCIAS EM ENGENHARIA DE TRANSPORTES.

Examinada por:

Prof. Hostilio Xavier Ratton Neto, Dr.

Prof. Márcio de Almeida D'Agosto, D. Sc.

Prof. Glaydston Mattos Ribeiro, D. Sc.

RIO DE JANEIRO, RJ - BRASIL
NOVEMBRO DE 2009

Kaiser, Maxwell Scardini.

Aplicação de Metaheurística Híbrida na Resolução do Problema Dial-a-Ride/ Maxwell Scardini Kaiser. – Rio de Janeiro: UFRJ/COPPE, 2009.

XII, 70 p.: il.; 29,7 cm.

Orientador: Hostilio Xavier Ratton Neto.

Dissertação (mestrado) – UFRJ/ COPPE/ Programa de Engenharia de Transportes.

Referencias Bibliográficas: p. 67-70.

1. Transporte. 2. Otimização - Transporte. 3. Cluster search. 4. Dial-a-ride. I. Ratton Neto, Hostilio Xavier. II. Universidade do Rio de Janeiro, COPPE. Programa de Engenharia de Transportes. III. Título.

Dedicatória

*A Deus seja Honra, Glória e Louvor para todo sempre.
À Família pelo amor, apoio e confiança.*

Agradecimentos

A Deus Soberano e Fiel toda Honra, Glória e Louvor.

A minha Família pelo apoio incondicional que nos momentos mais difíceis não me desampararam. Poder contar com vocês tem me permitido alçar vôos altos.

Em especial à minha esposa pelo amor, carinho e, sobretudo compreensão com a falta de tempo, nunca me deixando pensar em desistir dos meus objetivos. Ninguém conquista algo sozinho, sonhamos juntos e hoje podemos compartilhar mais esta vitória.

Aos meus tios e primos pelo apoio e carinho a mim dispensado, especialmente à família do primo Otávio pela acolhida.

Aos amigos que direta ou indiretamente contribuíram para a concretização de mais esta etapa, jamais vou esquecê-los.

Ao Prof. Hostilio pela orientação, confiança e apoio.

Aos Professores componentes da banca examinadora, em especial o Prof. Glaydston do CEUNES/UFES pela dedicação, apoio e esforços em auxiliar-me.

Aos companheiros de trabalho, em especial Cel. Dias, Saul, Karina, Elizabeth e Beatriz pelo apoio e esforços despendidos em ajudar-me na conclusão deste trabalho.

Aos amigos (as) da secretaria do PET, em especial Helena, Jane e Jorge Alberto, pela força e dedicação a mim dispensadas.

À CAPES pelo apoio financeiro para realização deste trabalho.

Resumo da Dissertação apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências (M. Sc.).

APLICAÇÃO DE METAHEURÍSTICA HÍBRIDA NA RESOLUÇÃO DO
PROBLEMA *DIAL-A-RIDE*

Maxwell Scardini Kaiser

Novembro/2009

Orientador: Hostilio Xavier Ratton Neto

Programa: Engenharia de Transportes

O objetivo deste estudo é analisar a aplicação de uma metaheurística híbrida na resolução do problema *dial-a-ride* – *DARP*. Estes problemas têm como característica principal o transporte de pessoas que apresentam alguma deficiência física que reduza a mobilidade. O *DARP* é considerado um problema com função multiobjetivo, pois visa garantir o atendimento da demanda ao menor custo possível, sujeito a manutenção da qualidade dos serviços prestados ao usuário. Diversos pesquisadores têm dispensado grandes esforços na tentativa de construir um modelo que represente o *DARP* próximo da realidade. Estudos realizados demonstram que o *DARP* pertence à classe dos problemas *NP-difícil*. Os modelos e métodos exatos utilizados para resolvê-lo, tem se mostrado ineficientes para os casos com mais de 55 requisições de transportes. Diante disto, este trabalho visa a aplicação de métodos heurísticos capaz de resolver o *DARP* em tempo computacional aceitável. Estes métodos não garantem que a solução encontrada seja a ótima, porém permite que soluções boas sejam encontradas com menor esforço computacional. Neste trabalho utiliza-se o método conhecido como *Cluster Search* (CS) e a metaheurística *Simulated Annealing* (SA), na resolução do *DARP*. Ao comparar os resultados obtidos através de métodos tradicionalmente utilizados como Algoritmo Genético, Busca Tabu, *Simulated Annealing* (SA) e outros, o CS-SA apresentou resultados semelhantes e, na maioria dos casos, melhores. Assim, é possível afirmar que o algoritmo CS se consolida como uma ferramenta importante na resolução de problemas computacionais de alta complexidade como é o caso do *DARP*.

Abstract of Dissertation presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Master of Science (M.Sc.)

APPLICATION OF HYBRID METAHEURISTIC IN TACKLING DIAL-A-RIDE
PROBLEM

Maxwell Scardini Kaiser

November/2009

Advisor: Hostilio Xavier Ratton Neto

Department: Transportation Engineering

The objective of this study is analyze the hybrid metaheuristic application in dial-a-ride problem – DARP resolution. These problems have as main feature the people transportation who present some physically disable that reduce their mobility. DARP is considered a problem with multiobjective function, because it guarantees the demand attendance in the lesser possible cost, with a quality's maintenance of services to clients. Many researchers have dispensed efforts trying to construct a model that represents the DARP more next of reality. Studies demonstrate that DARP NP-hard. The models and exact methods utilized to solve it, have indicated inefficiency to cases with more than 55 transportation requests. This study aims the application of heuristic methods able to solve it. These methods do not ensure that a solution found is the best one, however they find good solutions with less computational effort. This dissertation proposes a hybrid method know as Cluster Search (CS) and Simulated Annealing (SA) for solving DARP based on two metaheuristics. Comparing the results obtained against traditional methods such as Genetic Algorithm, Tabu Search, Simulated Annealing and others, the CS-SA presented similar results and, in many situations, better results. Therefore, it is possible to assert that CS algorithm a useful approach for solving high complexity problems, such as DARP.

SUMÁRIO

LISTA DE FIGURAS	ix
LISTA DE TABELAS	x
LISTA DE QUADROS	xi
LISTA DE ABREVIATURAS E SIGLAS	xii
1 Introdução.....	1
1.1 Visão geral do DARP	1
1.2 Objetivos e Justificativas.....	2
1.3 Organização e Método da Pesquisa.....	3
1.4 Estrutura da Dissertação	4
2 Revisão Bibliográfica	6
2.1 Problemas de Roteirização de Veículos	6
2.1.1 Classificação dos Problemas de Roteirização	10
2.2 Problemas de Roteirização e Programação de Veículos	12
2.3 Problemas Dial a Ride – DARP	15
2.4 Modelagem e Resoluções do DARP	19
2.5 Algoritmos propostos para resolução do DARP	21
3 Formulações e Modelos.....	28
3.1 Formulação Geral do DARP	28
3.2 Modelo proposto por Mauri e Lorena (2009).....	31
3.3 Métodos Heurísticos.....	35
3.3.1 A Metaheurística Cluster Search	36
3.3.2 A Metaheurística Simulated Annealing.....	40
3.4 Cluster Search e Simulated Annealing aplicados ao DARP	41
3.4.1 Representação do DARP	42
3.4.2 Detalhamento da Implementação do CS-SA	43
3.5 Considerações Finais	50
4 Avaliação dos Experimentos Computacionais	51
4.1 Descrição dos Resultados	53
4.2 Considerações Finais	62
5 Conclusões e Recomendações.....	64
Referências Bibliográficas.....	67

LISTA DE FIGURAS

Figura 2.1 – Variações do DARP	16
Figura 2.2 – Serviço de Coleta de um passageiro	18
Figura 2.3 – Troca Intrarotas.	23
Figura 3.1 – Fluxograma do CS	38
Figura 3.2 – Convergência dos métodos CS e SA.....	39
Figura 3.3 – Estrutura dos pontos.....	42
Figura 3.4 – Rota de atendimento.....	42
Figura 3.5 – Lista de rotas	42
Figura 3.6 – Heurística de distribuição.....	44
Figura 3.7 – Heurística de programação.....	44
Figura 3.8 – Algoritmo Simulated Annealing utilizado no CS	46
Figura 3.9 – Movimento reordenar rota	47
Figura 3.10 – Movimento realocar pontos	47
Figura 3.11 – Movimento trocar pontos	48
Figura 3.12 – Algoritmo EXECUTAR-CS	49
Figura 3.13 – Algoritmo utilizado na atualização dos centros de clusters	49
Figura 3.14 – Busca local	50
Figura 4.1 – Estrutura das instâncias utilizadas.....	52
Figura 4.2 – Tempo processamento R1a a R10a.....	61
Figura 4.3 – Tempo processamento R1b a R10b	62

LISTA DE TABELAS

Tabela 4.1 – Comparação do CS-SA com o SA proposto por Mauri e Lorena (2009)...	54
Tabela 4.2 – Melhores soluções obtidas pelo CS-SA.....	55
Tabela 4.4 – Tempo de duração das rotas	57
Tabela 4.5 – Tempo médio de viagem	58
Tabela 4.6 – Tempo médio de espera.....	59
Tabela 4.7 – Tempos de Processamento.....	60

LISTA DE QUADROS

Quadro 2.1 – Caracterização dos PRPVs	13
---	----

LISTA DE ABREVIATURAS E SIGLAS

Ceturb-GV: Companhia de Transportes Urbanos da Grande Vitória

CS: *Cluster Search*

DARP: *Dial a Ride Problem*

ECS: *Evolutionary Clustering Search*

GRASP: *Greedy Randomized Adaptive Search Procedures*

PMR: Pessoa com Mobilidade Reduzida

PRVs: Problema de Roteirização de Veículos

PRPVs: Problema de Roteirização e Programação de Veículos

SA: *Simulated Annealing*

VNS: *Variable Neighborhood Search*

1 Introdução

Este capítulo apresenta uma visão geral do problema, o objetivo e as justificativas mais relevantes que motivaram esta pesquisa. Estudos similares ao apresentado nesse trabalho têm buscado resolver e apresentar novas soluções que visam garantir a mobilidade e a acessibilidade dos portadores de deficiências, ao menor custo possível.

Pesquisas realizadas apontam para um crescimento da demanda por serviços do tipo *dial-a-ride*. Este crescimento, não se restringe apenas ao transporte de pessoas com deficiência física, conforme chegou-se a convencionar no início dos anos 80, mas principalmente, nos países desenvolvidos estes serviços tem sido estendidos as mais diversas áreas como atendimento domiciliar por equipes médicas, atendimento a idosos, usuários de entorpecentes, etc.

1.1 Visão geral do *DARP*

Dentre a classificação dos problemas de roteirização e programação de veículos, aqueles que envolvem o transporte de passageiros de seus locais de origem aos seus respectivos destinos, são conhecidos genericamente na literatura como problemas do tipo “*dial-a-ride*” (Znamensky e Cunha, 1999).

O problema de *Dial-a-Ride* (*DARP*, do inglês, *Dial-a-Ride Problem*), consiste no desenvolvimento de rotas e escalas de veículos para transportar diversos usuários, os quais especificam requisições de embarque e desembarque (coleta e entrega) entre locais de origem e destino específicos, (Cordeau e Laporte, 2007). O objetivo principal desse processo é planejar um conjunto de rotas para alguns veículos, com “custo mínimo”, capaz de acomodar o maior número possível de usuários, e sempre obedecendo a um conjunto de restrições (Mauri e Lorena, 2009).

Considerando, especificamente o caso do transporte de pessoas portadoras de necessidades especiais, um ponto de embarque (coleta) corresponde ao endereço onde um determinado cliente deve ser buscado, e o ponto de desembarque (entrega) associado corresponde ao endereço onde o mesmo cliente deve ser entregue. Cada ponto de embarque e seu respectivo ponto de desembarque, juntamente com suas respectivas “janelas de tempo”, formam uma requisição de transporte de um cliente (Mauri e Lorena, 2009).

O *DARP* pode ser considerado com vários objetivos distintos, como por exemplo:

- Minimizar os custos operacionais sujeito à satisfação de toda a demanda;
- Maximizar a satisfação da demanda, sujeito à disponibilidade dos veículos; ou
- A combinação dos itens anteriores, no qual, o objetivo primário é a busca pelo equilíbrio entre custo operacional e qualidade do serviço prestado.

1.2 Objetivos e Justificativas

Os Objetivos desta pesquisa estão definidos da seguinte forma:

- Estudar um modelo que represente o *DARP* o mais próximo da realidade;
- Implementar em linguagem computacional um algoritmo que resolva o modelo em tempo aceitável;
- Avaliar os resultados obtidos e compará-los com os disponíveis na literatura.

Os problemas relacionados aos sistemas de transportes tem sido frequentemente, objeto de discussões, as quais têm identificado necessidade de intervenções para que melhorias sejam alcançadas, de forma a facilitar a vida da população, principalmente, às localizadas nos grandes centros urbanos. Alguns desses problemas podem ser resolvidos de maneira simples, outras, porém, necessitam de estudos mais aprofundados, devido a aspectos como alta complexidade, alto consumo de recursos, etc. Dentre os problemas complexos que ainda necessitam ser resolvidos, estão os problemas de roteirização e programação de veículos.

Considerando que o nível de demanda por serviços do tipo *dial-a-ride* tem aumentado, e isto ocorre não só no Brasil, mas em todo o mundo, faz-se necessário o desenvolvimento de novos métodos automatizados capazes de encontrar soluções que levem a otimização do sistema, garantindo a satisfação dos usuários, a um custo total mínimo.

Neste estudo, toma-se por base outras pesquisas que consideram o mesmo problema e que procuram resolvê-lo da mesma forma proposta nesta pesquisa. Estas análises diminuem o risco, e esforços desnecessários, na compreensão de modelos, que se mostraram pouco eficientes na resolução destes problemas. A implementação de novos métodos heurísticos pode apresentar soluções alternativas para instâncias de problemas

conhecidos, e assim, contribuir para o aumento do nível de serviço, bem como a diminuição dos recursos consumidos no funcionamento adequado do sistema.

Outro ponto importante que merece destaque é o caráter social do serviço, que se destina a atender a parcela da população com dificuldades de se deslocar por meios de transporte convencionais, ou seja, pessoas portadoras de necessidades especiais, sobretudo aquelas pertencentes a classes sociais desprovidas de recursos próprios para executar deslocamentos para acessar serviços básicos como saúde, educação, lazer, etc. Sabe-se também que a população brasileira está em processo de envelhecimento, o que torna necessário a criação de serviços de coleta e entrega para o atendimento dessas pessoas, garantindo-lhes os direitos básicos assegurados pela Constituição Federal Brasileira, como serviços de saúde, cultura e lazer.

1.3 Organização e Método da Pesquisa

A fim de levantar os principais aspectos que envolvem um estudo da importância do *DARP*, existe a necessidade de se fazer um levantamento dos modelos e métodos propostos para resolver tais problemas.

Os modelos e métodos encontrados na literatura consideram diversos aspectos de acordo com a função objetivo e as características mais importantes que se deseja minimizar. Por isso, os levantamentos feitos servem para fundamentar o modelo, bem como os métodos heurísticos aplicados na resolução do *DARP*. Como este é um problema com função multiobjetivo, a definição do método adequado é muito importante, pois a escolha equivocada pode levar a resultados ruins ou até mesmo tornar inviável a utilização do mesmo, na resolução do problema.

O desenvolvimento desta pesquisa considera os seguintes aspectos principais:

- **Análise e levantamento do problema:** Consiste na análise do problema, os métodos de resolução e seus respectivos resultados. Desta forma é possível conhecer os fundamentos do problema, suas características mais relevantes e quais os métodos que melhor se adaptam ao problema, possibilitando uma melhor escolha do método a ser implementado.
- **Definição do Modelo e do Método Implementado:** As informações obtidas na etapa anterior são fundamentais para a escolha do modelo e métodos que serão utilizados. O modelo analisado neste trabalho é o que representa o *DARP* de

forma aproximada aos problemas reais, pois o mesmo considerada os aspectos fundamentais do problema.

- **Implementação Computacional:** Após a conceituação do problema e a escolha dos métodos, dá-se a fase de implementação computacional. Nesta pesquisa optou-se por utilizar a linguagem computacional C++, a fim de avaliar o comportamento do algoritmo *Cluster Search* (CS) e da metaheurística *Simulated Annealing* (SA).
- **Avaliações e Comparações:** Após o término da implementação é feita uma comparação entre os resultados obtidos com os de outros pesquisadores, permitindo avaliar a qualidade dos mesmos, bem como o grau de adaptação dos métodos na resolução do modelo escolhido para representar *DARP*. Esta é uma forma utilizada, frequentemente, na medição da qualidade das soluções obtidas, uma vez que, por meio do uso de modelos e métodos que procuram resolver tais problemas de forma exata, não foi possível encontrar uma solução ótima para as instâncias utilizadas.

1.4 Estrutura da Dissertação

Este trabalho está dividido e organizado da seguinte forma:

O Capítulo 1 consiste da introdução em que é apresentado o problema objeto desta pesquisa de forma geral e sucinta, os objetivos e as justificativas e a metodologia empregada no desenvolvimento deste estudo.

No Capítulo 2, encontra-se a revisão bibliográfica e os esforços para contextualizar o *DARP* dentro dos problemas classificados como problemas de Roteirização de Veículos. Ainda neste capítulo é feito uma revisão dos métodos mais utilizados para resolvê-lo.

No Capítulo 3 é apresentado a formulação geral do problema, os aspectos que o caracterizam e o modelo matemático proposto por Mauri e Lorena (2009) para resolvê-lo. Encontra-se também uma descrição dos métodos heurísticos *Cluster Search* e *Simulated Annealing* e as adaptações necessárias para a resolução do problema.

No Capítulo são apresentados os resultados dos experimentos computacionais realizados neste trabalho, bem como uma comparação com outros três trabalhos encontrados na literatura pesquisada, sendo estas as mais importantes publicações sobre o assunto, que procuram resolver as instâncias propostas por Cordeau e Laporte (2003b)

e, que também são utilizadas nesta pesquisa, como maneira de avaliar os métodos *Cluster Search* e *Simulated Annealing* – CS-SA implementados.

O Capítulo 5 apresenta as conclusões resultantes das análises feitas nos resultados obtidos com a aplicação dos métodos CS-SA. Tais conclusões permitiram ainda esboçar um conjunto de observações que seguem como recomendações aos pesquisadores que desejam, a partir deste trabalho, realizar novas pesquisas sobre este tema de sublimar importância, tendo como aspecto principal o caráter social do mesmo.

2 Revisão Bibliográfica

Este capítulo é resultado das análises efetuadas nos estudos mais importantes e de elevado destaque sobre o *DARP*. Ao analisar os trabalhos citados neste capítulo, tornou-se possível adquirir conhecimento suficiente para o desenvolvimento do trabalho proposto, garantindo o alcance do objetivo principal do mesmo. Nesta busca por informações relevantes que caracterizam o referido problema em questão, foi possível chegar ao estado atual da arte, conhecer os modelos e formulações propostas para representá-lo e os métodos que melhor se adaptam na resolução de cada modelo, dado que as características destes podem inviabilizar a utilização de modelos genéricos.

Tendo como base as análises apresentadas neste capítulo, o *DARP* pode ser resolvido de várias formas, mas para que os resultados sejam satisfatórios é necessário uma observação criteriosa dos aspectos que caracterizam o mesmo e como estão representados dentro do universo dos problemas classificados como de roteirização de veículos.

2.1 Problemas de Roteirização de Veículos

Segundo Novaes (1989), o principal problema logístico associado a transporte, é o de roteirização de veículos, conhecido na literatura como o Problema de Roteirização de Veículos (PRVs).

De maneira generalizada, o problema de roteirização de veículos consiste em definir um conjunto de pontos, que devem ser atendidos sequencialmente, visando à minimização dos custos totais de atendimento. Esse conjunto de pontos, também chamado de roteiro, é definido a partir de uma base ou depósito, garantindo o atendimento de cada ponto e sua respectiva demanda exatamente uma vez de forma que a capacidade do veículo não seja excedida em nenhum ponto do roteiro (Cordeau e Laporte, 2003b). Entretanto, na maioria dos casos, este modelo básico não condiz com a realidade das organizações e clientes envolvidos, existindo ainda, outras restrições operacionais que na prática são indispensáveis ao se analisar um problema de tamanha importância. Um exemplo destes tipos de restrições é encontrado no caso em que cada cliente pode ter seu horário específico para atendimento, ou ainda, não só receber como também, enviar produtos, caso a empresa trabalhe com coleta e entrega de mercadorias. Além disso, deve-se levar

em consideração o tempo de carga ou descarga nos pontos de coleta e/ou entrega, dentre outros.

Os problemas de roteirização de veículos tem sido objeto de estudo em diversas áreas de pesquisa e sua aplicação aparece em diversos contextos, tais como:

- Entrega de mercadorias em estabelecimentos como bares e restaurantes;
- Reabastecimento de caixas eletrônicos;
- Abastecimento e transporte de combustíveis;
- Coleta de óleo residual em restaurantes para produção de Biodiesel;
- Serviços de entrega de produtos adquiridos pela internet como livros, eletrodomésticos, computadores, etc.;
- Captação de doações para atendimento a carentes e/ou vítimas de catástrofes;
- Transporte de pacientes entre unidades (centros de tratamento e hospitais) ou busca de pacientes em casa para tratamento em unidades hospitalares;
- Coleta de Lixo, material para exames laboratoriais, medição de serviços como energia, água, etc.;
- Entrega de encomendas expressas (correios);
- Transporte de pessoas portadoras de necessidades especiais por meio da utilização de veículos de baixa capacidade, como vans, veículos adaptados, etc.

Com o aumento das restrições operacionais que implicam em aumento da complexidade do problema, encontrar intuitivamente uma boa solução na programação e execução da distribuição física de produtos, serviços e/ou pessoas, torna-se uma tarefa cada vez mais difícil. É neste contexto que diversas pesquisas têm sido realizadas, a fim de se alcançar um modelo que melhor represente estes problemas, bem como a construção de uma ferramenta flexível, que seja eficiente no planejamento dos roteiros, de forma que garanta, a otimização da operação dos sistemas de distribuição.

Desde a segunda metade do século XX, os problemas de roteirização de veículos vêm sendo objetos de estudos em diversas áreas como Ciência da Computação, Engenharia de Produção, Engenharia de Transportes, etc. Isto se deve a dois fatores principais: o primeiro diz respeito à importância da resolução do problema para o setor operacional

das organizações e, o segundo, por representar uma parcela significativa dos custos de distribuição.

No Brasil, diversos pesquisadores têm concentrado esforços na resolução dos PRVs, características como o tipo de serviço (porta a porta) e aliados ao fato do País possuir uma matriz de transportes caracterizada pelo modo rodoviário, são as principais justificativas para estes estudos. Como este modo é o responsável pelo transporte de pessoas e dos mais diversos tipos de produtos, os PRVs possuem grande importância para o desenvolvimento das empresas e cidades.

Dentre os principais fatores que aumentam a importância dos problemas de roteirização, encontram-se os seguintes:

- O aumento dos custos de transporte e o fato dos mesmos corresponderem a maior parte dos custos logísticos;
- A mudança nos cenários econômicos nacional, principalmente, após o plano real com a estabilidade da economia;
- A globalização e economia de mercado (aumento da competitividade);
- As exigências dos clientes em relação a prazos, datas e horários de entrega têm aumentado;
- As restrições e dificuldades de encontrar locais de estacionamento, principalmente, nos grandes centros urbanos, bem como o crescimento das cidades, aliadas à crescente taxa de motorização;
- O avanço dos sistemas da computacionais;
- A popularização dos computadores e ferramentas de apoio à tomada de decisão;
- O desenvolvimento dos modelos de programação matemática voltados para otimização de sistemas de transporte, como modelos em rede, *branch-and-bound*, relaxações lagrangianas, heurísticas e metaheurísticas, etc.

Alguns dos fatores destacados acima foram impulsionados, pelo advento da internet e o surgimento do comércio eletrônico. Estes fatores têm provocado mudanças no perfil dos consumidores, não só pela possibilidade de comprar sem sair de casa, como também receber os produtos num prazo curto. A internet também tem possibilitado ainda a

utilização de roteirizadores *online*, para a construção de roteiros, que tornem os deslocamentos mais eficientes.

Mesmo depois de mais de meio século estudando os problemas de roteirização de veículos, ainda não foi possível estabelecer, em muitos casos, um modelo capaz de solucioná-los de forma ótima. A alta complexidade destes problemas os elevou a classe *NP-Difícil*, ou seja, o tempo de resolução aumenta de forma exponencial à medida que se aumenta o número de pontos de atendimento (requisições). Segundo Cunha (2000), alguns dos fatores mais importantes que dificultam a construção destes modelos e, conseqüentemente a aplicação de métodos para resolvê-los, são:

- Falta de conhecimento do problema, que implica em escolhas erradas (métodos, *software*, etc.);
- Dados que não representam a realidade;
- Soluções podem variar mesmo para dados e problemas similares; e
- Problemas com grande número de restrições.

Segundo Brejon e Belfiore (2006), existem duas características próprias que podem tornar mais difícil a resolução dos problemas de roteirização de veículos. Ainda segundo os autores, estas dificuldades podem resultar em erros na formulação e resolução dos PRVS e, por conseguinte, a qualidade dos resultados:

- **Escolha do problema a resolver:** qual o problema? Identificar o problema a ser resolvido, bem como seus objetivos, é essencial para a construção de um modelo capaz de solucioná-lo. Muitas vezes, ao se tentar resolver um problema perde-se muito tempo procurando a resposta certa ao invés de procurar a questão certa a responder.
- **Nível de conhecimento:** A escassez de informação provoca incertezas ou falta de confiabilidade da solução, por outro lado, o excesso de informação pode provocar um aumento do tempo de resolução, ou até mesmo tornar inviável a representação matemática do problema, ou ainda, inviabilizar a resolução do mesmo por meio da utilização de métodos computacionais. Assim sendo, o equilíbrio entre *escassez* e *excesso* é determinado pelo nível de conhecimento que se tem acerca do problema em análise.

As buscas por novas soluções computacionais para diversos tipos de problemas, não devem ser justificadas apenas pela rapidez com que se obtêm os resultados, pois, resultados rápidos não são sinônimos de resultados bons. No caso específico dos problemas de roteirização de veículos, Brejon e Belfiore (2006) apresentaram resultados de uma análise dos benefícios alcançados com a utilização de ferramentas computacionais, aplicadas especificamente na resolução destes problemas. Dentre os benefícios citados destacam-se:

- A obtenção de rotas mais curtas e mais rápidas;
- Diminuição do número de veículos utilizados;
- Redução dos custos operacionais;
- Maximização da utilização do veículo (capacidade);
- Aumento do nível de serviço ao cliente; e,
- Melhorias na capacidade de tomada de decisão.

O fato dos problemas de roteirização possuírem objetivos comuns, não significa dizer que os métodos de resolução são aplicados de forma trivial, pois existem algumas características específicas que os transformam em problemas diferentes, levando à construção de modelos diversos que implicam na utilização de métodos distintos de para resolvê-los. Por isso, a construção de modelos que representem tais problemas, geralmente, segue alguns passos comuns e são formuladas dentro de uma determinada classificação preliminar.

2.1.1 Classificação dos Problemas de Roteirização

Segundo Bodin *et al.* (1983), os problemas de roteirização estão classificados em problemas de roteirização pura e problemas de roteirização e programação combinados. Nos problemas de roteirização pura, o objetivo principal é minimizar a distância total percorrida, considerando para isso, apenas as condicionantes espaciais, ou seja, não são avaliadas aspectos como as restrições temporais e a precedência de serviços (coleta e/ou entregas). Já os problemas de roteirização e programação combinados, consideram as características espaciais, temporais e de precedência, de forma que, a solução encontrada, seja a que represente a minimização dos custos, respeitando sempre os aspectos que caracterizam o problema em estudo. Geralmente, os PRVs são modelados

como um problema de grafos em que as demandas podem estar concentradas nos nós (vértices, pontos) e/ou nos arcos (arestas), porém não necessariamente em todos os arcos, pois assim, o mesmo seria caracterizado como um problema do carteiro chinês.

Existem outras classificações, mas para Cunha (2000), a classificação feita por Bodin *et al* (1983) é considerada uma das mais importantes, pois avalia os principais tipos de problemas de roteirização de veículos.

Podem ser encontradas ainda, outras formas de classificar os PRVs. Uma dessas é por meio da hierarquização dos parâmetros de acordo com a função objetivo proposta no modelo. Dentre esses parâmetros, encontram-se os seguintes:

- Clientes: estes possuem demanda que pode ser determinística ou estocástica. Quanto à localização, podem estar em um ponto ou arco. Além das restrições espaciais, os clientes podem ter ainda, restrições temporais, preferência, etc.;
- Frota: Pode ser homogênea e heterogênea. Outra característica importante é o tamanho da frota;
- Restrições aos veículos: Quanto aos veículos, deve ser considerada a capacidade (peso e volume), tipo de carga e de operação. Funcionários ou tripulação: envolvem dados como carga horária, intervalos para refeições, horas extras, etc.
- Tipo de operação: quanto à operação, podem ser somente serviços de coleta ou entrega, ou ainda, um combinado dos dois. Em caso de serviço combinado acrescenta-se a restrição de precedência, que neste caso é em função da demanda.
- Tipo de carga: pode ser única, mista. O tipo de carga possibilita o operador determinar qual o tipo de veículo será necessário para atendimento da demanda.

Pode-se perceber que diante do grande número de parâmetros que compõem um problema de roteirização, existe uma enorme dificuldade na modelagem de tais problemas, pois estes englobam vários parâmetros essenciais, que interferem de maneira considerável na função objetivo, de tal forma que, até mesmo uma hierarquização dessas características não é uma tarefa simples. Sendo assim, para se ter um modelo que represente o problema e, conseqüentemente, uma boa solução, é necessário conhecer

bem os fundamentos do mesmo, a fim de que este o modelo proposto represente a realidade da forma mais simples possível. Assim, dentre os objetivos comuns nos problemas de roteirização destacam-se os seguintes:

- Minimização dos custos variáveis;
- Minimização dos custos fixos;
- Minimização da soma de custos fixos e variáveis;
- Minimização do tempo de duração das rotas;
- Minimização do número de veículos necessários para atendimento da demanda;
- Maximização do nível de serviço e/ou satisfação e/ou prioridades dos clientes;
- Balanceamento de rotas;
- Dimensionamento de frota; e,
- Minimizar a utilização de frota própria ou fretada.

2.2 Problemas de Roteirização e Programação de Veículos

Os problemas classificados como de roteirização e programação de veículos – *PRPVs*, possuem aspectos que os tornam diferenciados, quanto ao número de restrições. Como citada anteriormente, além das restrições consideradas nos problemas de roteirização pura, os *PRPVs* consideram ainda aspectos temporais como, horário de atendimento, precedência de tarefas, etc.

Uma classificação dos problemas de roteirização e programação de veículos bastante difundida na literatura foi proposta por Ronen (1988). Nesta classificação são considerados aspectos operacionais e quais os objetivos devem ser alcançados, respeitando sempre as características do problema em análise. A seguir são destacadas algumas dessas características de acordo com o tipo de problema:

- Problemas relativos ao transporte de passageiros: programação de linhas de ônibus; serviços de táxi; serviços de transporte de pessoas com mobilidade reduzida, geralmente, pessoas portadoras de necessidades especiais e idosas,

conhecidos como problema *dial-a-ride*; transporte de escolares por ônibus, vans e outros;

- Problemas de prestação de serviços: roteirização e programação de equipes de reparos ou de serviços públicos, tais como: coleta de lixo (regular e seletiva), entrega postal, leitura de parquímetros, instalação e conserto de telefone fixo, atendimento a chamadas domiciliares, como manutenção de máquinas, tubulações, etc.; Faria (2005) apresentou uma formulação para reprogramação de rotas em tempo real para o caso de atendimento as solicitações de reparos por clientes de uma operadora de telefonia fixa. Para validar a formulação foi aplicado um método heurístico que possibilitou o alcance de bons resultados.
- Problemas relativos ao transporte de carga (coleta e distribuição). Pode ser considerado o mais simples de resolver, pois, a aplicação de novos conceitos na análise do problema, apresentam significativas reduções no número de parâmetros que precisam ser considerados. Como exemplo, pode-se estabelecer horários para entregar os produtos em clientes onde exista a dificuldade de acesso em horário comercial. Desta forma, após essas definições, são criadas rotas específicas para o atendimento a esses pontos, previamente estabelecidos, facilitando a aplicação de janelas de tempo mais “relaxadas”, simplificando a modelagem matemática do problema. Isto se deve, principalmente, a sua importância para o setor operacional das organizações e pelo custo de distribuição representar uma parcela significativa do custo total.

Quadro 2.1 – Caracterização dos PRPVs

Características	Parâmetros
<ul style="list-style-type: none"> • Frota 	<p>Tamanho: Um ou mais veículos.</p> <p>Tipo: Homogênea, heterogênea e veículos adaptados (transporte de deficientes).</p> <p>Compatibilidade: tipo de veículo e o tipo de produto a ser transportado.</p> <p>Localização: uma ou mais garagens.</p>

<ul style="list-style-type: none"> • Clientes 	<p>Demanda: determinística, estocástica.</p> <p>Localização; nós, arcos, combinação arcos e nós.</p>
<ul style="list-style-type: none"> • Veículos 	<p>Tipo de carga: única ou múltiplas.</p> <p>Capacidade: peso, volume.</p> <p>Tipo de operação: coleta, entrega e a combinação de serviços de coleta e entrega.</p>
<ul style="list-style-type: none"> • Condições Temporais 	<p>Duração da rota, não deve exceder um tempo máximo permitido.</p> <p>Tempo de atendimento.</p> <p>Horário de coleta e entrega.</p> <p>Janelas de tempo rígidas ou não.</p>
<ul style="list-style-type: none"> • Atendimento 	<p>Tipo de requisição: somente coletas, entregas, coletas de retorno.</p> <p>Tamanho da demanda: um único produto ou múltiplos produtos.</p> <p>Atendimento parcial ou total da demanda.</p> <p>Status: conhecimento da demanda a priori e existência de incertezas na demanda.</p> <p>Frequência: Necessidade de programação de visitas periódicas com frequências definidas.</p> <p>Nível de prioridade: baixa ou alta prioridade de atendimentos.</p>

Um dos principais problemas de roteirização e programação de veículos, que apresentam diversas características demonstradas no Quadro 2.1, é o que envolve o transporte de pessoas portadoras de necessidades especiais e/ou pessoas idosas, conhecido como problema *dial-a-ride* ou do inglês *Dial-a-Ride Problem - DARP*. Pode ser intuitivo dizer que, os problemas *dial-a-ride* são simplesmente, problemas de

roteirização e programação. Mas, apesar do *DARP* apresentar características similares ao serviço de coleta e entrega, existem outros aspectos que os tornam um pouco mais complexos. Como o *DARP* é caracterizado pelo transporte em que as cargas são pessoas e não objetos, a qualidade dos serviços prestados é um fator essencial que necessita ser considerado ao se tentar modelar tal problema. Faria (2005) apresenta um estudo de caso que é um *PRPVs*, que trata do atendimento de clientes de telefonia fixa, mesmo sendo um serviço que considera vários aspectos semelhantes ao *DARP*, este não pode ser classificado como um problema do tipo *dial-a-ride*.

Pode-se perceber que, o grande número de restrições consideradas nos *PRPVs*, dificulta a construção de um modelo matemático, que represente fielmente, o que ocorre em casos reais. Entretanto, dependendo da função objetivo, é possível organizar essas características de forma que apenas algumas sejam consideradas, facilitando com isso a implementação de modelos matemáticos, capazes de representar problemas reais. Isto ocorre, por exemplo, no transporte escolar, onde os transportados são pessoas e os parâmetros que envolvem tempo, são de fundamental importância, para o nível de serviço prestado.

2.3 Problemas *Dial a Ride* – *DARP*

Segundo Znamensky e Cunha (1999) os problemas modelados como problemas de roteirização e programação de veículos – *PRPVs* podem ser aplicados em diversos tipos de situações. Dentre essas destacam-se aquelas que envolvem o transporte de passageiros de seus locais de origem aos seus destinos, são conhecidos genericamente na literatura, como problemas *dial-a-ride* ou simplesmente *DARP*,. Geralmente, estes problemas envolvem o transporte porta-a-porta de pessoas idosas e/ou portadoras de necessidades especiais. Ao contrário dos serviços de táxis convencionais, em que cada passageiro é atendido de maneira individual, no *DARP*, várias pessoas, com diferentes origens e destinos, podem ser transportadas ao mesmo tempo pelo veículo, ou seja, pode-se ter em determinado ponto do roteiro, várias pessoas com destinos diferentes e ainda, uma demanda heterogênea. O termo *dial-a-ride* pelo qual são conhecidos decorre da possibilidade dos serviços serem solicitados por telefone.

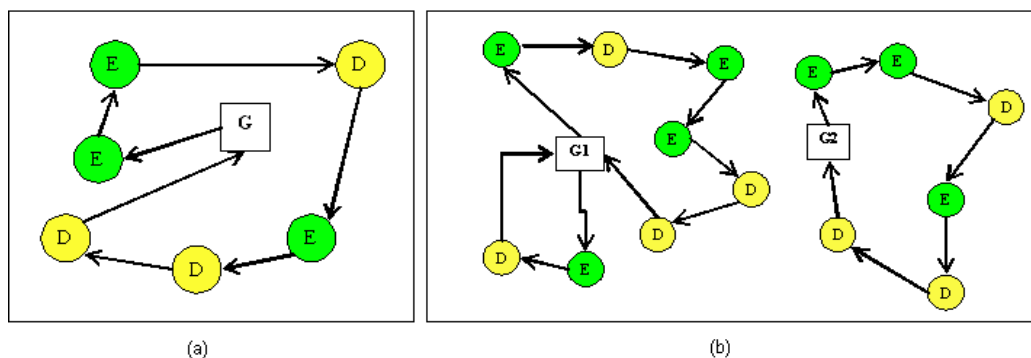
O *Dial-a-Ride Problem* – *DARP* consiste no desenvolvimento de rotas e escalas de veículos para transportar diversos usuários, os quais especificam requisições de

embarque e desembarque (coleta e entrega) entre locais de origem e destino específicos, (Cordeau e Laporte, 2007).

No caso específico do transporte de pessoas com mobilidade reduzida (PMRs) ou portadoras de necessidades especiais, um ponto de embarque (coleta) corresponde ao endereço onde uma determinada PMR deve ser apanhada, e o ponto de desembarque (entrega) associado, corresponde ao endereço onde a mesma PMR deve ser entregue. Cada ponto de embarque e seu respectivo ponto de desembarque, juntamente com suas respectivas “janelas de tempo”, formam uma requisição de transporte de um cliente (Mauri e Lorena, 2009).

A maioria dos estudos relacionados ao *DARP* admite que, a frota disponível para atendimento das requisições de transportes é homogênea e está localizada em uma única garagem, porém, isto não é uma regra geral em situações reais, principalmente em regiões geográficas de grandes dimensões. Existem casos com várias garagens e dependendo do tipo de serviço prestado, a frota pode ser heterogênea, como é o caso, por exemplo, das organizações que prestam serviços a diversos tipos de pessoas com mobilidade reduzida, em que às vezes é necessário transportar não só cliente, mas também, sua cadeira de rodas (Mauri e Lorena, 2009).

A Figura 2.1 apresenta essas possibilidades de disposição dos veículos: a parte (a) mostra o caso com garagem única, representada por (G) e a parte (b) garagens múltiplas, representadas por G1 e G2. Nos dois casos a requisições de embarque são representados pelos círculos verdes com a letra *E*, e as de desembarque pelos amarelos com a letra *D*.



Fonte: Adaptado de Mauri (2008).

Figura 2.1 – Variações do DARP

Existem variações dos PRPVs, para as quais ainda não foram encontradas uma solução ótima, como é o caso do *DARP*. Isto se deve, principalmente, ao aumento da complexidade do problema à medida que se aumenta o número de restrições do mesmo.

A utilização de técnicas variadas, como por exemplo, as técnicas de inteligência artificial, têm sido ferramentas imprescindíveis na construção de soluções boas, mas, não ótimas para muitos casos.

Os problemas conhecidos como *DARP* podem ser subdivididos em problemas estáticos e dinâmicos. No primeiro caso, todas as requisições de transporte são conhecidas a priori, enquanto que no segundo caso, são consideradas requisições feitas ao longo do dia (normalmente por telefone). Neste último caso, no início do dia conhece-se apenas alguns pontos de atendimento, sendo que as novas requisições surgidas durante o dia deverão ser alocadas às rotas e atendidas pelos veículos que as operam, ou seja, as rotas são ajustadas em tempo real de acordo com a inserção de novas demandas, mas estes casos, não são muito comuns na prática (Cordeau e Laporte, 2003b). Apesar dos modelos dinâmicos serem uma alternativa que permite uma interatividade maior com o usuário, a aplicação destes modelos em problemas reais torna-se praticamente inviável, devido o grande custo computacional para resolvê-los, bem como a dificuldade de serem implementados, tornando assim, o uso dos modelos estáticos mais intensos, pois são mais simples de serem aplicados na prática.

Por ser o estudo do *DARP* voltado para o atendimento de pessoas portadoras de necessidades especiais, geralmente, associado a algum fator que reduz mobilidade, o nível de serviço torna-se um aspecto de grande importância. Esta é a principal característica que justifica o estudo do *DARP* de maneira distinta dos demais problemas de roteirização e programação de veículos, pois ao transportar pessoas, surge a necessidade de se garantir o melhor nível de serviço. Uma forma de avaliar a qualidade dos serviços prestados é estabelecer um limite máximo permitido para as “inconveniências” aos usuários. A garantia de um melhor nível de serviço caminha de forma inversa ao objetivo principal de todo e qualquer prestador de serviço, que é o menor custo total possível.

O estudo do *DARP* tem sido justificado nos países da Europa e América do Norte por ações que visam, não só implantar serviços para atendimento da crescente demanda, mas, revisar a gestão dos sistemas em funcionamento, a fim de garantir o bem estar das pessoas portadoras de necessidades especiais e dos idosos, etc. Outro fator importante que tem contribuído para estas pesquisas é a necessidade de ajustes nos gastos por parte das organizações, ou seja, existe um crescimento da demanda, que precisa ser atendida a um custo mínimo. Cordeau e Laporte (2003a) apresentam ainda, dois fatores principais

que motivam o crescimento de tais serviços. O primeiro considera o envelhecimento da população e, o segundo, a preocupação dos governos com a implantação de novos serviços de atenção a saúde domiciliar e assistência às pessoas em processo de desintoxicação (usuários de drogas).

No Brasil, este tipo de transporte se restringe a pouquíssimas cidades, algumas como São José dos Campos – SP, Sorocaba – SP, São Paulo – SP e na região da Grande Vitória – ES, por exemplo, já prestam o serviço à população que necessita de tais atendimentos. Ao optarem por um serviço diferenciado, para atender esta parcela da população, os governos garantem bem estar e integração social.

No caso do Espírito Santo, o serviço é denominado “Serviço Especial Mão na Roda”, implantado pelo Governo do Estado através da Ceturb-GV (Companhia de Transportes Urbanos da Grande Vitória). O objetivo principal do projeto é oferecer um atendimento diferenciado de transporte coletivo urbano, para pessoas portadoras de necessidades especiais, que fazem uso de cadeiras de rodas. A implantação deste serviço se deve às dificuldades que esses cidadãos têm de se locomoverem por meios do transporte convencional, garantindo assim, a mobilidade dos chamados “cadeirantes”, promovendo a inclusão social nos seus compromissos profissionais, médicos, de educação ou de lazer. A Figura 2.2 mostra o embarque de usuário do “Serviço Especial Mão na Roda”.



Fonte: Ceturb-GV (2009).

Figura 2.2 – Serviço de Coleta de um passageiro

Outro exemplo muito interessante deste tipo de problema é apresentado por Melachrinoudis *et. al.* (2007). Segundo os autores, nos Estados Unidos o aumento do número de usuários de drogas e álcool é um problema crescente, e muitas organizações

de saúde tem se dedicado diariamente a prestar serviços de atendimento a pessoas afetadas por tais problemas. Ainda segundo os autores, dois fatores fundamentais serviram de base para este estudo, a crise econômica e a necessidade de ampliação do número de atendimentos. No estudo apresentado pelos autores o conjunto de requisições de transportes representa viagens programadas para desintoxicação, atendimento médico/psicológico, entrevistas em centros de recuperação, etc. Neste estudo de caso, o levantamento encontrou uma descentralização do sistema de transporte, entre os centros de tratamento, que resultavam em ineficiências no sistema, como duplicação de viagens, horários de atendimento desordenados e, encaminhamento manual dos procedimentos. Estas ineficiências produziam crescentes custos de transporte, viagens inoportunas, longo tempo de espera por atendimento, tempo extra de viagem, tempo ocioso e subutilização da capacidade dos veículos. Desta forma, o principal objetivo do trabalho foi desenvolver um modelo matemático, capaz de minimizar custos de transportes e os inconvenientes para os clientes, considerando todas as requisições, obedecendo sempre um conjunto de restrições. Assim, os autores utilizaram um modelo *dial-a-ride*, a fim de minimizar os custos com transportes e os inconvenientes aos usuários.

Outro aspecto importante que merece ser destacado é o processo de envelhecimento da população brasileira. Desta forma, surge a necessidade de implantação de serviços de coleta e entrega para o atendimento á essas pessoas, garantindo com isso o acesso aos serviços básicos essenciais, garantidos constitucionalmente.

2.4 Modelagem e Resoluções do *DARP*

Analisando os modelos que representam o *DARP*, vê-se que os mesmos demonstram algumas características semelhantes, pois, procuram resolvê-lo de maneira que possibilite a redução dos custos com transporte, satisfazendo um conjunto de restrições. Outras características podem ser encontradas nos problemas classificados como *DARP*, de forma que os mesmos sejam transformados em problemas com função multiobjetivo (Mauri e Lorena, 2009). Dentre esses aspectos, destacam-se:

- Custos operacionais, sujeito a satisfação de toda a demanda;
- Satisfação da demanda, sujeito a disponibilidade dos veículos;
- Distância total percorrida;
- Número de veículos utilizados para o atendimento das requisições;

- Tempo de duração das rotas (rota não deve exceder o tempo máximo);
- Tempo de permanência dos clientes (usuários) dentro dos veículos (tempo de viagem não deve exceder tempo máximo);
- Tempo de espera nos locais de embarque e desembarque (tempo de espera não deve exceder tempo máximo);
- Capacidade do veículo não pode ser excedida em nenhum momento;
- Início do serviço deve respeitar os intervalos pré-estabelecidos.

Podem existir ainda, outras características conhecidas como restrições de atendimentos, isto se deve pela necessidade das requisições estarem divididas em coletas (embarque) e entregas (desembarque), gerando assim, a restrição de precedência uma vez que não é possível desembarcar alguém que não foi embarcado. Conforme Znamensky e Cunha (1999), tais restrições não estão presentes em problemas de roteirização que envolvem a coleta ou distribuição física de produtos, tornando inviável a aplicação de métodos de resolução conhecidos na literatura, como o método das economias de Clark e Wright (1964).

Para Cordeau e Laporte (2007), uma tendência comum na construção dos modelos é permitir que os usuários, definam suas janelas de tempo de atendimento. Mas este procedimento dificulta substancialmente, a aplicação de métodos de resolução, especialmente, nos casos onde as janelas definidas são estreitas. Por isso, consideram que os modelos *DARP* podem ter dois aspectos gerais:

- Minimizar os custos, sujeito a plena satisfação da demanda;
- Maximizar a satisfação da demanda, sujeito a disponibilidade dos veículos.

Em tempos de avanços tecnológicos parece contraditório dizer que, para muitos problemas de roteirização, ainda não se conhece soluções ótimas e, dentre estes problemas está o *DARP*. Isto se deve, a fatores como modelos sem equilíbrio, quanto aos requisitos a serem minimizados, e ainda, a complexidade implícita em tais problemas. No caso do *DARP*, estudos mostram que problemas com apenas 60 pontos de atendimento, não se conhece uma solução ótima. Diante de tais dificuldades e pelas características presentes nos problemas reais, pode-se afirmar que, é uma tarefa bastante complexa, resolvê-los de forma que se alcance a otimalidade.

Diante do exposto anteriormente, vê-se que formular matematicamente o problema não é uma tarefa simples e requer conhecimentos aprofundados sobre o problema, bem como, sensibilidade para extrair as características mais importantes. Outros fatores, relacionados com o que se deseja minimizar na função objetivo, exercem forte influência na construção de modelos, para resolver deste tipo de problema. Como exemplo, pode-se definir um modelo que priorize a minimização dos custos operacionais em detrimento dos benefícios. Mas no caso do *DARP*, esta não seria uma boa alternativa de solução, pois o nível de serviço é muito importante para o funcionamento adequado do sistema.

No capítulo 3 que trata o desenvolvimento do trabalho, será descrito o modelo matemático elaborado por Cordeau e Laporte (2006), bem como as adaptações feitas por Mauri e Lorena (2009), que visam introduzir uma melhoria no nível de serviço nas soluções apresentadas.

2.5 Algoritmos propostos para resolução do *DARP*

Geralmente, os PRPVs encontrados na literatura seguem uma classificação de acordo com os métodos de resolução, que podem ser exatos e heurísticos. Como pode ser visto nos trabalhos de Cordeau e Laporte (2003b) e em Mauri e Lorena (2009), os métodos exatos apresentam sérias dificuldades quanto ao tempo de processamento despendido na tentativa de resolver problemas de ordem média (50 requisições), dificultando com isso sua aplicação na resolução de problemas reais, uma vez que, nesses casos os números de requisições são bem maiores que isso. Métodos heurísticos são mais rápidos que os métodos exatos, porém não garantem que a solução ótima seja alcançada. Dentre os métodos heurísticos mais utilizados para resolver estes problemas, destacam-se Algoritmo Genético (AG), *Simulated Annealing* (SA), *Greedy Randomized Adaptive Search Procedures* (GRASP), Colônia de Formigas (CF), Busca Tabu (BT) e Inserção Paralela (IP).

Desrosiers *et al.* (1986) apresentaram um algoritmo de tempo polinomial para resolver um problema em que os atendimentos possuíam janelas de tempo e, deveriam ser realizados por um único veículo. Este método foi baseado em programação dinâmica e de forma sistemática consiste em eliminar estados incompatíveis que violam restrições de capacidade do veículo, tempo e precedência. O algoritmo se mostrou

computacionalmente eficiente para resolver problemas relativamente pequenos, ou seja, capaz de resolver instâncias do *DARP* com até 40 requisições.

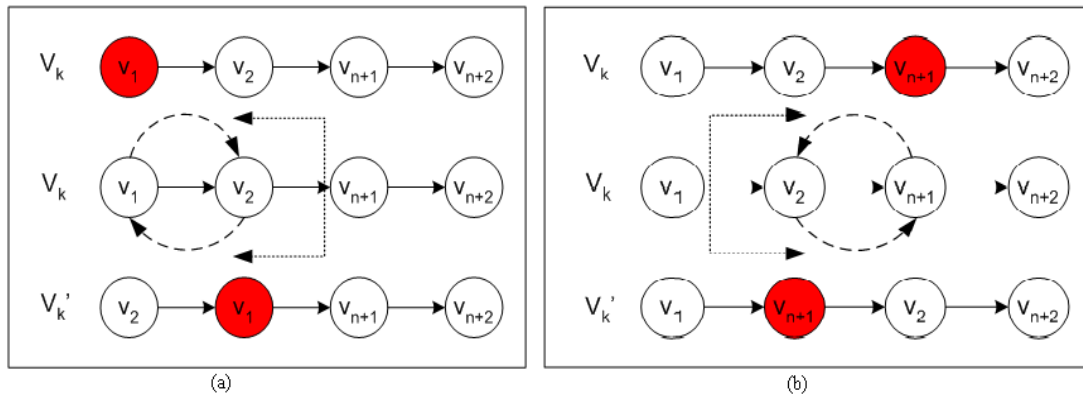
Jaw *et al.* (1986) propuseram uma heurística de inserção paralela para problemas *dial-a-ride*, com múltiplos veículos e restrições de janelas de tempo. Neste trabalho, o problema foi considerado de forma bem próxima da realidade, ou seja, com múltiplos veículos e com restrições de janelas de tempo. Os autores apresentaram uma heurística denominada *Heurística de Inserção Paralela*, cuja ideia principal consiste em ordenar as solicitações de atendimento por um critério de horário, e inserí-las, segundo a ordenação adotada, nas rotas que resultem em menor aumento de custo. Este método foi testado com sucesso para uma base de dados de aproximadamente 2600 clientes e 20 veículos. É importante destacar que essas instâncias representavam o problema para vários dias de operação, ou seja, nem todos os clientes solicitavam atendimento no mesmo dia.

Na tentativa de resolver o *DARP* e encontrar uma solução ótima, Dumas *et al.* (1991) desenvolveram um método exato, baseado no algoritmo de *Geração de Colunas*. O problema considerado era composto por frota homogênea e restrições de janela de tempo. O método proposto neste trabalho mostrou ser eficiente para resolver problemas com no máximo 55 solicitações de atendimento.

Madsen *et al.* (1995) introduziram modificações no método desenvolvido por Jaw *et al.* (1986), adotando um critério de dificuldade para ordenar as solicitações. O objetivo principal deste critério é garantir prioridade a solicitações com restrições mais rígidas, tornando o procedimento de inserção mais eficiente, pois nos passos iniciais da heurística, os veículos alocados nas rotas estão mais vazios, e há um número maior de alternativas de inserções viáveis.

Toth e Vigo (1996) apresentaram uma heurística de inserção paralela para o problema com múltiplos veículos e restrições de janela de tempo, incluindo um método de melhoria de busca local. Neste modelo, o problema foi dividido em duas categorias. A primeira, chamada de troca *intra-rotas*, consiste na tentativa de melhoria que se dá pela alteração na sequência em que as requisições de transporte serão atendidas. Já a segunda, chamada de troca *inter-rotas*, considera a possibilidade de trocar requisições de transporte entre rotas distintas. A Figura 2.3 apresenta um exemplo da troca *intra-rotas*. Segundo os autores, o método foi testado com sucesso em um conjunto de cerca de 1500 solicitações procedente de um caso real, na cidade de Bologna - Itália. Ao

aprimorar o método de melhoria com a inclusão do método de busca local, derivado do conhecido método da Busca Tabu, os autores conseguiram significativas melhorias, quando comparados com os resultados obtidos na versão original do projeto (Toth e Vigo, 1997).



Fonte: Mauri (2008).

Figura 2.3 – Troca Intrarotas.

Znamensky e Cunha (1999) abordaram o *DARP* utilizando uma adaptação da Heurística de Inserção Paralela proposta por Madsen *et al.* (1995), acompanhada de uma fase de melhoria das rotas obtidas, semelhante a apresentada por Toth e Vigo (1996). Vale destacar aqui, uma característica muito importante deste trabalho: o mesmo foi substanciado com o estudo de caso real, extraído da operação do ATENDE, serviço de transporte de deficientes gerido pela São Paulo Transporte S/A. Este serviço é voltado, especificamente, para o atendimento de idosos e portadores de necessidades especiais por veículos de pequena capacidade (peruas ou vans). Essas instâncias correspondem a um dia típico do mês de novembro de 1998, totalizando 349 requisições de transporte (clientes) e 84 veículos disponíveis, localizados em 47 garagens distribuídas no município de São Paulo.

A partir do ano de 1998 a aplicação de metaheurísticas tem crescido significativamente, pois estes métodos, apesar de não garantirem a obtenção da solução ótima, possibilitam a inserção de várias restrições, de forma que o aumento da complexidade do problema seja amenizado. Metaheurísticas são métodos de melhorias que trabalham a partir de uma solução inicial, tentando encontrar uma solução ótima. Em muitos casos, esses métodos são empregados com a finalidade de reduzir o esforço computacional necessário para resolver problemas de alta complexidade, como é o caso do *DARP*. As metaheurísticas mais encontradas na literatura para resolver estes problemas são Algoritmos Genéticos, Busca Tabu, Colônia de Formigas e *Simulated Annealing*.

Baugh *et al.* (1998) apresentaram um estudo no qual o *DARP* é resolvido utilizando a metaheurística conhecida como *Simulated Annealing*. Neste caso, aplicou-se a técnica conhecida como “agrupar primeiro” e “roteirizar depois”, ou seja, as requisições de atendimento são primeiramente organizadas em *clusters* e, em seguida, os roteiros e as programações são resolvidos para cada um desses *clusters*, separadamente. A fase de agrupamento é realizada pelo *Simulated Annealing* e, a fase seguinte, com a aplicação da heurística do vizinho mais próximo. Os resultados obtidos são baseados em um conjunto de dados de um problema real com 300 clientes. Os autores destacam ainda, que os resultados encontrados são muito próximos do ótimo global.

Jih *et al.* (2002) utilizaram Algoritmo Genético para resolver um problema de coleta e entrega, com janelas de tempo e apenas um veículo. Esse algoritmo permite a exploração de soluções inválidas durante a busca, neste caso, uma rota é definida por um cromossomo que representa a sequência de locais a serem atendidos pelo veículo. A função objetivo considera apenas os custos de viagem do veículo e as violações nas restrições. Nessa abordagem, são utilizadas instâncias geradas aleatoriamente com até 100 clientes, sendo o tempo de processamento necessário para resolvê-las de aproximadamente 38 minutos.

Uma abordagem que utiliza a metaheurística Busca Tabu para resolver o *DARP* é apresentada por Cordeau e Laporte (2003b). Neste caso, o problema é tratado de forma estática, com diversos veículos, frota homogênea e garagem única. Outro detalhe deste estudo é a comparação de três métodos heurísticos, *P1*, *P2* e *P3*, que, juntamente com a Busca Tabu, são utilizados para resolver o problema. O primeiro (*P1*) visa apenas a minimização das violações nas janelas de tempo, enquanto o segundo (*P2*), além de procurar minimizar essas violações, busca também a minimização do tempo de duração rotas. Já o terceiro (*P3*) considera os mesmos aspectos que os anteriores, porém objetiva ainda a redução nos tempos de permanência dos clientes no interior dos veículos. Os resultados obtidos mostram que *P1* e *P2*, por serem mais “simples”, são mais rápidos que *P3*, porém as soluções geradas por *P3* são significativamente melhores. Todos os resultados são obtidos a partir de instâncias geradas aleatoriamente, baseadas em informações cedidas pela *Montreal Transit Commission* (MTC) de Montreal - Canadá. Essas instâncias variam de 24 a 144 clientes. Além disso, eles apresentam outros resultados obtidos para instâncias de problemas reais (com 200 e 295 clientes) cedidas por uma transportadora dinamarquesa.

Diana e Dessouky (2004) apresentaram uma nova heurística para a versão estática do *DARP* com janelas de tempo. Neste estudo desenvolveu-se uma rota de inicialização por meio da introdução de um pedido inicial para cada um dos veículos, considerando os efeitos espaciais e temporais como pesos. No passo seguinte utilizou-se a heurística de inserção paralela com pesos, para o restante dos pedidos que não foram inseridos na inicialização das rotas, em vez de classificar os pedidos por determinados critérios, por exemplo, menor prazo de entrega. Da mesma forma que as heurísticas de inserção clássicas, esta inserção acumula uma matriz de custo incremental para cada um dos pedidos não atribuído e quando atribuída a cada uma das rotas existentes. O peso dos custos é uma medida de potencial dificuldade, se um pedido não é imediatamente atribuído, então é calculado o peso para cada pedido, e o algoritmo busca o pedido com maior peso e procura inseri-lo nos horários atuais. Todo o procedimento é repetido até que todos os pedidos sejam inseridos. Neste estudo, considerou-se frota sem restrições de capacidade, o que dificulta a aplicação desta técnica em problemas reais, pois tais características estão presentes nestes casos. O sucesso do método comprovou-se pela aplicação do algoritmo para um estudo de caso real, com 1.000 solicitações de transporte, resultando em uma redução de 8% da frota em comparação com o clássico modelo de inserção sequencial, proposto por Solomon (1987). No entanto, para 100 requisições, o estudo aponta que a melhoria na qualidade da solução usando a heurística de inserção por peso, é marginal.

Attanasio *et al.* (2004) apresentaram um estudo onde procuraram resolver o *DARP* de maneira dinâmica, utilizando computação paralela por meio da implementação de heurísticas paralelas a partir dos conceitos aplicados no método da Busca Tabu. Tal estudo foi motivado pela boa adaptação do método da Busca Tabu na resolução dos problemas estáticos, realizados por Cordeau e Laporte (2003b). Os experimentos computacionais apresentados por Attanasio *et al.* (2004) demonstraram que a utilização de computação paralela pode proporcionar grandes benefícios para a resolução em tempo real de problemas correlacionados ao *DARP*.

Cordeau (2006) apresenta uma abordagem exata para resolver o problema utilizando o algoritmo *Branch-and-Cut*, garantindo assim a obtenção da solução ótima para o problema que foi tratado de forma estática, com múltiplos veículos, frota homogênea e garagem única. Os resultados foram obtidos a partir de instâncias geradas aleatoriamente com no máximo 32 clientes. Mauri e Lorena (2009) apresentam um

modelo matemático para resolver o problema de forma exata, entretanto, o software CPLEX (ferramenta bastante utilizada em modelos de otimização) não foi capaz de resolver em 60 minutos instâncias que combinam 24 a 144 requisições de atendimento e 3 a 13 veículos, porém ao aplicar uma metaheurística foi encontrada uma solução boa com tempo inferior a 3 minutos. Como em problemas reais o número de requisições normalmente são bem superiores às utilizadas nos experimentos, torna-se justificável então, a implementação de novos modelos e métodos capazes de resolver esses problemas em tempos computacionais menores.

Uma aplicação do método de programação inteira mista conhecido como *Brach and Bound* e da heurística Busca Tabu é apresentada por Melachinoudis *et al.* (2007). Ao estudar um problema real, de uma organização de saúde nos Estados Unidos, os autores procuraram resolvê-lo considerando duas funções objetivos. A primeira propõe a redução do custo total de transportes, enquanto que a segunda está relacionada com a qualidade do serviço, pois busca a minimização da insatisfação dos usuários, através da diminuição do tempo de espera por atendimento, menor tempo entre atendimentos, etc. Devido às características do problema, os autores encontraram dificuldades para aplicar o método exato *Brach and Bound* na resolução de instâncias maiores. Isto justificou a utilização do método heurístico da Busca Tabu para solucionar o problema.

Abordando o *DARP* de forma estática com múltiplos veículos, múltiplas garagens e frota heterogênea, Jorgensen *et al.* (2007) apresentaram um estudo que emprega Algoritmo Genético para resolvê-lo de forma semelhante ao que foi proposto por Baugh *et al.* (1998). Para encontrar a solução, os autores se valeram de uma forma clássica de tratar o problema, conhecida como método de duas fases, “agrupar primeiro” e “roteirizar depois”. Na primeira fase, o Algoritmo Genético é utilizado para agrupar (distribuir) os clientes nos veículos, ou seja, determinar quais clientes serão atendidos, por determinado veículo. Já na fase seguinte, ocorre a roteirização e a programação que é realizada de forma individual para cada veículo, por meio da heurística do vizinho mais próximo, sendo aplicado o mesmo conceito que Baugh *et al.* (1998). Os resultados são obtidos a partir das instâncias geradas aleatoriamente por Cordeau e Laporte (2003b).

Mauri e Lorena (2009) buscando aprimorar os estudos feitos por Cordeau e Laporte (2003b), compararam dois métodos para resolver o *DARP*. No primeiro caso utilizou-se a ferramenta CPLEX, na tentativa de resolver o modelo matemático de forma exata, não

sendo este capaz de encontrar uma solução ótima em 60 minutos, para instâncias que combinam 24 a 144 requisições de atendimento e 3 a 13 veículos. Assim, resolveram adotar uma combinação de heurísticas de programação e distribuição com a metaheurística *Simulated Annealing*, na tentativa de obterem êxito na resolução do problema. O modelo adotado pelos autores representa o problema de forma “relaxada”, ou seja, alguns itens considerados como restrições essenciais ao problema, tornam-se parte da função objetivo a ser resolvida, deixando assim, de serem vistas como simples restrições. Apesar desta abordagem permitir soluções inválidas, do ponto de vista computacional, isto é significativo, pois torna a resolução menos árdua. Os principais resultados obtidos por Mauri e Lorena (2009) estão descritos na Seção 4.1 deste trabalho, onde é realizada uma comparação com os encontrados pela aplicação dos métodos CS-SA proposto nesta pesquisa, para resolver o *DARP*.

Os estudos analisados, com exceção do apresentado por Attanasio *et al.* (2004), consideram o *DARP* de forma estática, isto pode ser justificado pela dificuldade de operação dos sistemas dinâmicos. Desde os anos 80 do século XX, têm sido apresentados estudos que procuram considerar o problema de maneira dinâmica, entretanto, tais pesquisas, na maioria dos casos não são aplicáveis em casos reais. Em algumas pesquisas que tratam o caso do *DARP* dinâmico, na prática parte apenas algumas requisições são avaliadas como dinâmica, ou seja, o problema é considerado dinâmico apenas nos aspectos macros.

3 Formulações e Modelos

Neste capítulo é apresentado a formulação geral do *DARP*, um modelo matemático que o representa e os métodos adotados neste trabalho, para resolvê-los. Como pôde ser visto no Capítulo 2 deste trabalho, existem diversas formas de resolver o *DARP*. Entretanto, os modelos que procuram representar o problema de forma exata têm apresentado dificuldades de serem resolvidos de forma ótima e, principalmente, serem aplicados na resolução de problemas reais. Com isso, pesquisadores têm procurado desenvolver modelos que os represente, de tal forma, que seja possível encontrar soluções aproximadas, por meio da utilização de métodos heurísticos, garantindo assim, a obtenção de soluções boas.

3.1 Formulação Geral do *DARP*

A formulação geral do *DARP* parte do princípio que existem n clientes (requisições de transporte) a serem atendidos por m veículos. Cada requisição de transporte é formada por um ponto específico de embarque i e um de desembarque $n+i$. Para representar o problema são definidos os seguintes conjuntos:

- K : conjunto dos veículos disponíveis ($|K| = m$).
- G^- : conjunto de garagens de origem.
- G^+ : conjunto de garagens de destino.
- P : conjunto dos locais de embarque.
- U : conjunto dos locais de desembarque.
- $N = \{G^- \cup P \cup U \cup G^+\}$: conjunto com todos os locais (pontos).

Um detalhamento desta representação mostra que para cada cliente i ($\forall i \in P$) existe uma demanda específica q_i que necessita ser atendida, ou seja, o número de assentos que o mesmo ocupará no veículo, outro dado importante é a definição de dois horários para atendimento. No primeiro é especificado um horário para embarque de um usuário no ponto de origem $[e_i, l_i]$ e, outro que ele gostaria de desembarcar no seu local destino $[e_{n+i}, l_{n+i}]$, estas informações são conhecidas como janelas de tempo. A carga q_i deve ser um valor positivo nos pontos de embarque e o mesmo valor, porém negativo, nos respectivos pontos de desembarque.

Para cada cliente i ($\forall i \in P$) existe um tempo máximo de viagem R_i associado, ou seja, o tempo máximo que o cliente poderá permanecer no interior do veículo. Para todo local i ($\forall i \in \{P \cup U\}$) é associado um tempo máximo de espera W_i , ou seja, o tempo máximo em que os veículos poderão ficar esperando até o início do serviço. Entende-se como serviço o embarque ou desembarque de um cliente em determinado local, sendo atribuído ao mesmo, um tempo adicional, caso seja necessário para a concretização do atendimento, representado por s_i . Após a definição dos pontos que representam as garagens e os pontos de requisições de transporte é possível obter as distâncias $d_{i,j}$, os tempos de duração das viagens $t_{i,j}$ e os custos de deslocamento $c_{i,j}$ entre os pontos i e j , onde $i, j \in N$ e $i \neq j$.

Para cada veículo k ($\forall k \in K$) existe uma capacidade Q_k (quantidade de assentos disponíveis), um tempo máximo de duração da rota associada T_k , e a indicação do início de sua rota em uma garagem específica G_k^- e termino da rota em outra garagem de destino G_k^+ , podendo ser a mesma ou não. Cada garagem possui suas próprias janelas de tempo para partida (garagem de origem) e chegada (garagem de destino).

Existe ainda um conjunto de restrições, geralmente chamadas de requisitos “essenciais”, que precisam ser consideradas na resolução do *DARP*. A obrigatoriedade da satisfação desses requisitos tem como finalidade principal, garantir soluções válidas, em casos de problemas práticos. Dentre esses requisitos destacam-se os seguintes:

- O tempo de duração da rota executada pelo veículo k ($\forall k \in K$) não deve exceder o tempo máximo permitido T'_k ;
- O tempo de viagem do cliente i ($\forall i \in P$) não deve exceder o tempo máximo de viagem permitido R'_i ;
- O tempo de espera no local i ($\forall i \in \{P \cup U\}$) não deve exceder o tempo máximo de espera permitido W'_i ;
- A capacidade Q'_k ($\forall k \in K$) dos veículos não pode ser excedida em nenhum ponto da rota; e,
- O início do serviço em todos os pontos de embarque i ($\forall i \in N$) deve estar dentro dos intervalos pré-estabelecidos $[e_i, l_i]$ e $[e_{n+i}, l_{n+i}]$;

Após todas essas definições tem-se que o objetivo do modelo é minimizar os custos operacionais e a “insatisfação” dos clientes, ou seja, minimizar os requisitos “não-

essenciais” do problema, satisfazendo sempre os requisitos essenciais considerados no problema em análise.

Após o levantamento de todas as informações descritas anteriormente, é possível então definir as variáveis de decisão, que servirão a todo processo de roteirização e programação dos veículos. Assim considera-se:

- A_i^k como sendo o horário de chegada no ponto i ($\forall i \in N$) pelo veículo k , sendo $A_i^k = 0$ se $i \in G^-$ e $A_i^k = D_{i-1}^k + t_{i-1,i}$ se $i \in \{P \cup U \cup G^+\}$;
- D_i^k representa o horário de partida do ponto i ($\forall i \in N$), novamente pelo veículo k ; sendo $D_i^k = 0$ se $i \in G^+$ e $D_i^k = B_i^k + s_i$ se $i \in \{P \cup U\}$ e $D_i^k = B_i^k$ se $i \in G^-$, tal que s_i é o tempo de realização do serviço;
- B_i^k representa o horário de início do serviço no local i ($\forall i \in N$), sendo $B_i^k = D_i^k$ se $i \in G^-$ e $B_i^k = \max\{e_i, A_i^k\}$ se $i \in \{P \cup U \cup G^+\}$;
- W_i^k representa o tempo antes do início do serviço no ponto i ($\forall i \in N$), sendo $W_i^k = 0$ se $i \in G^-$ e $W_i^k = B_i^k - A_i^k$ se $i \in \{P \cup U \cup G^+\}$;
- Q_i^k indica a carga (numero de assentos ocupados) do veículo k após o término do atendimento no ponto i ($\forall i \in N$), sendo $Q_i^k = 0$ se $i \in \{G^- \cup G^+\}$ e $Q_i^k = Q_{i-1}^k + q_i$ se $i \in \{P \cup U\}$; e,
- R_i^k como sendo o tempo de viagem do cliente i ($\forall i \in P$) no veículo k , sendo $R_i^k = B_{n+i}^k - D_i^k$.

Feito isso, considera então $x_{i,j}^k = 1$ se o veículo k se deslocar do ponto i para o ponto j e $x_{i,j}^k = 0$, caso contrário.

Considerando os aspectos mais relevantes que caracterizam o *DARP*, pesquisas desenvolvidas por especialistas em estudá-los, como pode ser visto em Cordeau (2006), Bergvinsdottir (2004) e Mauri e Lorena (2009), entre outros, demonstram que a construção de modelos matemáticos que representem o *DARP* de forma exata, não tem obtido resultados para casos que se assemelham a problemas reais. Desta forma, os

modelos matemáticos que permitem uma resolução utilizando métodos heurísticos, têm aumentado de maneira significativa entre os pesquisadores.

Visando aperfeiçoar ou mesmo amadurecer o conhecimento acerca dos problemas *dial-a-ride*, Cordeau (2006) apresenta um modelo matemático que “simplifica” a representação do problema. Uma das formas de se alcançar essa simplificação é por meio da redução do número de variáveis e restrições. Se por um lado facilita a implementação de métodos de resolução, por outro, essas simplificações podem inviabilizar a utilização de tal modelo em casos reais, uma vez que trata apenas de problemas com frota homogênea (todos os veículos possuem uma mesma capacidade), garagem única (todos os veículos começam e terminam sua rota na mesma garagem) e com custo fixo. Desta forma, não é possível alcançar o equilíbrio entre custos operacionais e as inconveniências dos usuários (clientes).

Visando introduzir aspectos que possibilitassem melhorias no nível de serviço, Mauri e Lorena (2009) apresentam um aperfeiçoamento do modelo proposto por Cordeau (2006). As alterações propostas pelos autores procura ainda garantir uma melhor adaptação do modelo a problemas reais. Como as instâncias desenvolvidas por Cordeau e Laporte (2003b), são originárias de casos reais, os resultados apresentados demonstram que o modelo matemático proposto por Mauri e Lorena (2009) considera o equilíbrio entre as inconveniências dos clientes e os custos operacionais, tais características possibilitam afirmar que o mesmo pode ser uma alternativa na resolução de problemas práticos.

3.2 Modelo proposto por Mauri e Lorena (2009)

Ao analisar os diversos modelos encontrados na literatura é possível afirmar que os mesmos apresentam grandes contribuições, na forma matemática de representar o *DARP*. Como estes problemas possuem alto grau de complexidade, encontrar um modelo que seja capaz de representá-lo, bem como ser aplicado na resolução de problemas reais, não é simples. Mesmo alguns desses modelos apresentando dificuldades de serem aplicados em casos práticos, pode-se afirmar que são excelentes pontos de partida para consolidar tais estudos.

No modelo matemático proposto por Mauri e Lorena (2009), o *DARP* é tratado de forma estática, ou seja, todas as requisições são conhecidas a priori, com múltiplos veículos, com frota heterogênea (cada veículo possui uma capacidade distinta) e

garagens múltiplas (cada veículo começa e termina sua rota em garagens específicas). Este modelo é derivado de outros modelos apresentados por Cordeau (2006), Cordeau e Laporte (2003a), Cordeau e Laporte (2003b) e Jorgensen *et al.*, 2007), e representa o problema de forma bastante aproximada com os problemas reais.

O modelo proposto por Mauri e Lorena (2009) é do tipo multiobjetivo, pois busca minimizar, concomitantemente, os custos operacionais e a “insatisfação” dos clientes, ou seja, minimizar os requisitos “não-essenciais” do problema. As principais diferenças entre os modelos propostos por Cordeau (2006) e Mauri e Lorena (2009) estão na forma de considerar os requisitos essenciais. Com o objetivo de facilitar a aplicação de métodos para resolver o modelo, Mauri e Lorena (2009) procuraram representar o problema de forma relaxada, ou seja, algumas restrições passaram a compor a função objetivo como pode ser visto nas equações 3.1 e 3.2. Essa consideração torna mais suave a obtenção de uma solução para o problema, porém permite encontrar soluções inválidas.

Como apresentado no início deste capítulo o modelo matemático proposto por Mauri e Lorena (2009) abrange os requisitos “essenciais” e os “não essenciais”. Entende-se por requisitos essenciais os apresentados na Seção 3.1. Já os requisitos “não essenciais” estão relacionados com a distância total percorrida pelos veículos, número de veículos utilizados no atendimento às requisições, tempo de duração das rotas, tempo de viagem dos clientes e tempo de espera nos locais de embarque e desembarque. A seguir é apresentado o modelo desenvolvido por Mauri e Lorena (2009):

Minimize:

$$w_0 \sum_{k \in K} \sum_{i \in N} \sum_{j \in N; j \neq i} (d_{i,j} x_{i,j}^k) + w_1 \sum_{k \in K} \sum_{j \in P} x_{g_k^-, j}^k + w_2 \sum_{k \in K} (B_{g_k^+} - D_{g_k^-}) + w_3 \sum_{i \in P} R_i + w_4 \sum_{i \in \{P \cup U\}} W_i + \quad (3.1)$$

$$\rho_0 \sum_{k \in K} \max\{0, (B_{g_k^+} - D_{g_k^-}) - T_k\} + \rho_1 \sum_{i \in P} \max\{0, R_i - R'_i\} + \rho_2 \sum_{i \in \{P \cup U\}} \max\{0, W_i - W'_i\} + \quad (3.2)$$

$$\rho_3 \sum_{k \in K} \max\{0, \left(Q_k \sum_{i \in \{P \cup U\}; j \in \{P \cup U\}; j \neq i; j \neq n-i} x_{i,j}^k \right) - Q_k\} + \rho_4 \sum_{i \in N} (\max\{0, e_i - B_i\} + \max\{0, B_i - l_i\}) \quad (3.3)$$

Sujeito a:

$$\sum_{j \in \{P \cup \{g_k^+\}\}} x_{g_k^-, j}^k = 1 \quad \forall k \in K \quad (3.4)$$

$$\sum_{j \in \{U \cup \{g_k^-\}\}} x_{g_k^+, j}^k = 1 \quad \forall k \in K \quad (3.5)$$

$$\sum_{k \in K} \sum_{j \in \{P \cup U\}; j \neq i} x_{i, j}^k = 1 \quad \forall i \in P \quad (3.6)$$

$$\sum_{j \in \{P \cup U\}; j \neq i} x_{i, j}^k - \sum_{j \in \{P \cup U \cup \{g_k^+\}\}; j \neq i; j \neq n+i} x_{n+i, j}^k = 0 \quad \forall k \in K; i \in P \quad (3.7)$$

$$\sum_{j \in \{P \cup U \cup \{g_k^-\}\}; j \neq i; j \neq n+i} x_{i, j}^k - \sum_{j \in \{P \cup U\}; j \neq i} x_{i, j}^k = 0 \quad \forall k \in K; i \in P \quad (3.8)$$

$$\sum_{j \in \{P \cup U\}; j \neq i} x_{i, j}^k - \sum_{j \in \{P \cup U \cup \{g_k^+\}\}; j \neq i; j \neq n+i} x_{i, j}^k = 0 \quad \forall k \in K; i \in U \quad (3.9)$$

$$B_j = (B_i + s_i + t_{i, j} + W_j) \sum_{k \in K} x_{i, j}^k \quad \forall i, j \in N; i \neq j \quad (3.10)$$

$$Q_j = (Q_i + q_j) \sum_{k \in K} x_{i, j}^k \quad \forall i, j \in N; i \neq j \quad (3.11)$$

$$A_i = B_i - W_i \quad \forall i \in \{P \cup U \cup G^+\} \quad (3.12)$$

$$D_i = B_i + s_i \quad \forall i \in \{P \cup U \cup G^-\} \quad (3.13)$$

$$R_i = B_{n+i} - D_i \quad \forall i \in P \quad (3.14)$$

$$A_{g_k^-} = D_{g_k^+} = Q_{g_k^-} = Q_{g_k^+} = W_{g_k^-} = 0 \quad \forall k \in K \quad (3.15)$$

$$A_i, W_i, B_i, D_i, Q_i \quad \text{irrestritas} \quad \forall i \in N \quad (3.16)$$

$$R_i \quad \text{irrestrita} \quad \forall i \in P \quad (3.17)$$

$$x_{i, j}^k \in \{0, 1\} \quad \forall k \in K; \forall i, j \in N; i \neq j \quad (3.18)$$

Como descrito anteriormente, o modelo proposto por Mauri e Lorena (2009) é multiobjetivo e, é composto de duas partes. A Equação 3.1 procura minimizar os

requisitos não-essenciais do problema. Já as Equações 3.2 e 3.3 visam minimizar as violações dos requisitos “essenciais”. Além disso, os requisitos não-essenciais são “penalizados” por um vetor de números inteiros positivos (pesos) $w = [w_0; w_1; w_2; w_3; w_4]$, e os essenciais com um vetor semelhante $\rho = [\rho_0; \rho_1; \rho_2; \rho_3; \rho_4]$.

A finalidade das restrições 3.4 e 3.5 é garantir que cada veículo partirá de sua garagem de origem e chegará a sua garagem de destino uma única vez, ou seja, cada rota iniciará em sua garagem de origem e terminará na respectiva garagem de destino.

A restrição 3.6 define que cada cliente será atendido uma única vez, ou seja, por apenas um veículo. A restrição 3.7 visa garantir que um ponto de embarque estará sempre na mesma rota que seu respectivo ponto de desembarque. As restrições 3.8 e 3.9 servem para limitar o uso, ou seja, garantir que o total de embarques seja igual ao total de desembarques.

A restrição 3.10 determina o horário de início do serviço e o tempo de espera em cada ponto de embarque e também o veículo que o atenderá cada requisição de transporte. Outro objetivo deste conjunto de restrições é garantir a eliminação de sub-rotas (*sub-tours*), presentes no caso do *DARP*, como pode ser visto em Cordeau (2006). A restrição 3.11 representa a carga dos veículos em cada ponto, enquanto as restrições 3.12, 3.13 e 3.14 garantem, respectivamente, um cálculo correto dos horários de chegada e partida nos pontos de origem e destino e dos tempos de viagem dos clientes. A restrição 3.15 “inicializa” algumas variáveis referentes às garagens. Finalmente, a restrição 3.18 serve apenas para garantir que as variáveis de decisão $x_{i,j}^k$ sejam binárias, enquanto as restrições 3.16 e 3.17 garantem que as demais variáveis sejam irrestritas.

Segundo Mauri e Lorena (2009), uma das vantagens deste modelo é que o mesmo simplifica a forma de representar o problema, por meio da redução no número de variáveis e restrições, juntamente, com o número de arcos do problema da mesma forma que o proposto por Cordeau (2006), de forma que o problema fica mais adequado à realidade.

Finalmente, deve ser feito aqui uma observação sobre o modelo matemático proposto por Mauri e Lorena (2009). Segundo os autores, tal modelo pode ser aprimorado e até mesmo adaptado, provavelmente, possibilitando a utilização de ferramentas computacionais disponíveis no mercado.

3.3 Métodos Heurísticos

Devido à complexidade do modelo matemático apresentado na Seção 3.2 escolher um método capaz de resolvê-lo não é uma tarefa simples. O grande número de variáveis e restrições presentes no *DARP* dificulta essa escolha, uma vez que a escolha de um método influencia diretamente na qualidade da solução. Conforme apresentado anteriormente, métodos exatos não têm sido eficientes na resolução de problemas reais, pois estão limitados a resolver problemas com um número bastante reduzido de requisições de transporte, inviabilizando a aplicação destes em problema reais. Assim sendo, a utilização de métodos heurísticos combinados com metaheurísticas aplicados ao *DARP* tem sido uma prática comum entre os pesquisadores.

Método Heurístico é um procedimento que se baseia na experiência ou julgamento para encontrar uma solução aceitável, ou seja, não significa dizer que, necessariamente, esta seja a melhor. A boa adaptação destes métodos na resolução de problemas como o *DARP*, tem sido um dos aspectos que mais influenciam a aplicação em diversas áreas de pesquisas como Inteligência Artificial, Redes de Computadores e Arquitetura de Computadores, Engenharia de Transportes, Pesquisa Operacional, etc. O uso de métodos heurísticos tem sido justificado por alguns fatores, como:

- Métodos exatos nem sempre são capazes de resolver determinados problemas ou não é possível modelá-lo de forma exata;
- Métodos exatos inviáveis de serem tratados computacionalmente;
- Métodos heurísticos são de fácil compreensão, quando comparado a outros; métodos, facilitando a implementação computacional;
- Alto custo computacional para resolver o problema de forma exata; e,
- Necessidade de uma resposta de boa qualidade em um tempo computacional reduzido.

Os métodos heurísticos são classificados como métodos construtivos e melhorativos. No primeiro caso, o algoritmo tem como finalidade gerar a solução inicial do problema, partindo de determinado ponto, que pode ser um *cluster*, por exemplo. Já no segundo o algoritmo visa determinar uma melhoria nas soluções encontradas por meio da aplicação de uma heurística qualquer. Um dos gargalos na utilização de métodos heurísticos deve-se a dificuldade de escapar de ótimos locais, surgindo assim, outra metodologia, chamada de metaheurística, composta por mecanismos que procuram

forçar o escape destes pontos de ótimos locais, permitindo a busca em regiões identificadas como, supostamente, mais promissoras.

Como o objetivo principal deste trabalho é a aplicação de métodos heurísticos na resolução do *DARP*, a definição de quais métodos poderiam ser aplicados, só foi possível após um amplo estudo sobre o comportamento de diversos métodos em outras pesquisas sobre o assunto. Dentre os métodos encontrados observou-se que o *Simulated Annealing*, utilizado por Mauri e Lorena (2009) apresenta excelentes resultados, para as instâncias propostas por Cordeau (2003).

3.3.1 A Metaheurística *Cluster Search*

Após pesquisar quais métodos poderiam ser avaliados na resolução do *DARP*, chegou-se a conclusão de que a utilização de uma metaheurística híbrida poderia alcançar resultados importantes, uma vez que a aplicação desta técnica em outros problemas, tem apresentado resultados bastante promissores, no que diz respeito a tempo de processamento.

A metaheurística híbrida escolhida é conhecida como *Cluster Search* e, é um método derivado do aperfeiçoamento do algoritmo de Busca Evolutiva por Agrupamentos (ECS, do inglês, *Evolutionary Clustering Search*), proposto por Oliveira e Lorena (2004). Esta técnica conhecida também como heurística híbrida, combina metaheurísticas com outras técnicas de otimização e, seu objetivo principal é detectar e explorar áreas supostamente promissoras através de métodos de busca local. Essas características possibilitam que a busca seja intensificada somente em áreas que tenham um grande potencial de melhoria da solução. Desta forma, problemas com dimensões cada vez maiores podem ser resolvidos de forma satisfatória e em tempo aceitável, resultando em menor consumo de recursos.

A partir deste método, Oliveira (2004) propôs algumas modificações com o intuito de generalizar o algoritmo responsável por gerar soluções para o processo de agrupamento e clarear as idéias acerca do método. Devido a esta generalização, o nome do método foi simplificado para Busca por Agrupamentos (CS, do inglês *Cluster Search*). Basicamente, as modificações visam deixar o método mais eficiente e simples de ser implementado.

O *ECS* se baseia no fato de que, ao longo do processo de busca de um algoritmo evolutivo, há uma maior concentração de indivíduos em regiões nas quais estão os

indivíduos mais bem avaliados. Sendo assim, um processo de agrupamento é executado iterativamente com um algoritmo evolutivo, contabilizando o número de indivíduos gerados em regiões do espaço de busca e identificando grupos de indivíduos com similaridades, que mereçam especial atenção. A ideia central do método é detectar regiões promissoras por mérito de frequência, baseado na quantidade de soluções geradas em cada região de busca. Uma intensificação da busca é realizada nas regiões promissoras tão logo sejam detectadas (Oliveira, 2004).

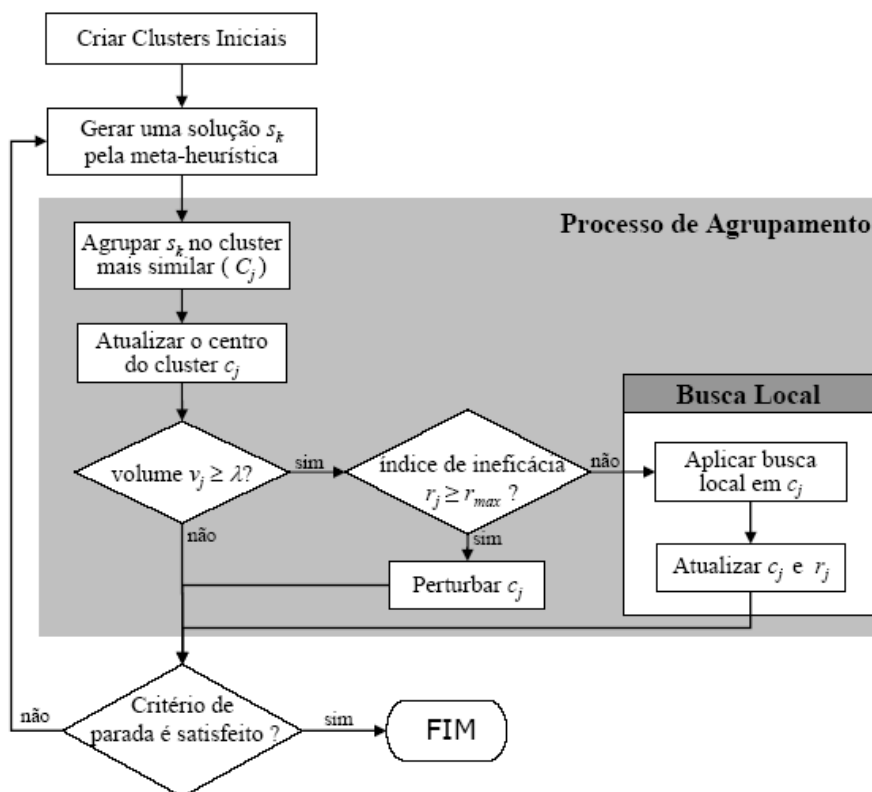
Já o *Cluster Search* procura dividir o espaço de busca e localizar regiões promissoras por meio do enquadramento desses em *clusters*. Um *cluster* pode ser definido por três atributos $C = \{c, v, r\}$. O centro c_i é uma solução que representa o *cluster* i , identificando a sua localização dentro do espaço de busca. Ao invés de armazenar todas as soluções agrupadas no *cluster*, apenas parte das informações destas soluções são atribuídas ao centro do *cluster*. O volume v_i é a quantidade de soluções agrupadas no *cluster* i . Um *cluster* se torna promissor quando o seu volume atingir certo limitante λ . O índice de ineficácia r_i é uma variável de controle para identificar se a busca local está ou não melhorando o centro do *cluster* i . O valor de r_i indica o número de vezes consecutivas que a busca local foi aplicada no *cluster* i e não melhorou a solução. Este atributo evita que a busca local seja executada em regiões ruins ou regiões que já tenham sido suficientemente exploradas por mais de r_{max} vezes.

Para que o algoritmo possa agrupar soluções em *clusters* é necessário definir alguma forma de medir a distância entre duas soluções. Sendo assim, uma função de medida de distância, $d(i, j)$, é definida, a priori, para calcular a distância entre duas soluções i e j , ou seja, a distância é um número positivo que é proporcional a distância entre as duas soluções.

O *Cluster Search* é um método iterativo que possui três componentes principais: uma metaheurística, um processo de agrupamento e uma heurística de busca local. A cada iteração do CS, uma solução S é gerada pela metaheurística e enviada para o processo de agrupamento. Essa solução é então agrupada no *cluster* mais similar C_j e o centro do *cluster* c_j é atualizado com informações contidas na nova solução agrupada, fazendo com que o centro se desloque no espaço de busca. Em seguida é analisado o volume v_j do *cluster*. Caso esse volume seja maior que um limitante λ , definido a priori, isso indica que algum padrão de solução está sendo predominantemente gerado pela metaheurística, e que esse *cluster* pode estar em uma região de busca promissora. Mas,

se a heurística de busca local não tiver obtido sucesso nas últimas r_{max} aplicações neste *cluster* (índice de ineficácia $r_j \geq r_{max}$) é aplicada uma perturbação aleatória no centro c_j , objetivando escapar de um possível ótimo local. Por outro lado, se $r_j < r_{max}$, uma busca local é aplicada no centro c_j analisando a vizinhança do *cluster*. Encerrado o processo de agrupamento, retorna-se para a metaheurística que irá gerar uma outra solução. O critério de parada do CS é definido pela metaheurística escolhida. A Figura 3.1 apresenta o fluxograma de execução do CS.

A principal vantagem do algoritmo *Cluster Search* se deve a aplicação de métodos busca local somente em regiões supostamente promissoras por meio do agrupamento de soluções que possuem certa similaridade. Assim o objetivo do CS é melhorar o processo de convergência da função objetivo associado a uma diminuição no esforço computacional em virtude do emprego mais racional dos métodos busca local.



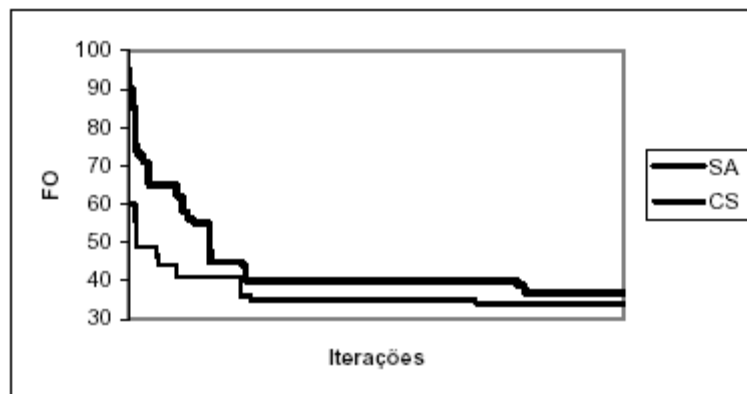
Fonte: Chaves (2009).

Figura 3.1 – Fluxograma do CS

Analisando a aplicação deste algoritmo em outros problemas, percebe-se que o mesmo vem apresentando ótimos resultados quando comparados a outros métodos heurísticos,

tradicionalmente, utilizados para resolver problemas de alta complexidade, como Algoritmo Genético, *Simulated Annealing*, Busca Tabu, etc.

Chaves *et al* (2007a) utilizou *Cluster Search* para o Problema de Balanceamento de trabalhadores em Linha de Produção e obteve resultados significativos, reduzindo substancialmente, os tempos de processamento computacional para a resolução do problema. Na Figura 3.2 encontra-se um gráfico comparando a convergência dos métodos *Cluster Search* e *Simulated Annealing* em um problema de otimização. Nele é possível visualizar que o primeiro converge para boas soluções de forma mais rápida que o segundo, sem o processo de agrupamento. Vê-se também que neste caso o *Cluster Search* encontrou uma solução de melhor qualidade que o *Simulated Annealing*.



Fonte: Chaves *et. al.* (2007a)

Figura 3.2 – Convergência dos métodos CS e SA

Chaves *et al.* (2007b) aplicaram *Cluster Search* (CS) para o problema de localização conhecido como *p*-medianas, com restrição de capacidade e conseguiu resultados expressivos, em relação aos métodos tradicionais utilizados para resolver tais problemas.

Chaves e Lorena (2008) utilizaram *Cluster Search* para o Problema do Caixeiro Viajante com Coleta de prêmios e o mesmo apresentou resultados muito promissores, sendo que para instâncias com 60 pontos de atendimento foi encontrada a solução ótima e, para os casos com um número de pontos superior, foram encontradas soluções em tempos computacionais razoáveis, comparados com os métodos que não utilizaram o conceito de agrupamento como no CS.

Os exemplos de aplicação do *Cluster Search* na resolução dos mais variados tipos de problema são os principais motivadores da utilização deste, na busca de uma melhoria

nos resultados do *DARP*, pois, os problemas resolvidos em outros trabalhos possuem a mesma classificação geral.

3.3.2 A Metaheurística *Simulated Annealing*

A metaheurística Recozimento Simulado ou *Simulated Annealing* (SA) foi proposta por Kirkpatrick *et al.* (1983), sendo uma das primeiras metaheurísticas relatadas na literatura. O SA procura simular o processo de recozimento físico, no qual um sólido é aquecido a uma alta temperatura (acima do seu ponto de fusão), e logo em seguida é realizado um processo lento e gradativo de arrefecimento, até que o ponto de solidificação seja atingido. Durante o recozimento, o material passa por vários estados possíveis. Se o resfriamento for suficientemente lento obtêm-se uma estrutura cristalina livre de imperfeições (estado de mínima energia).

Fazendo uma analogia com um problema de otimização combinatória, o nível de energia é a função objetivo do problema, um estado do sistema é uma solução viável, a temperatura é um parâmetro de controle e o estado de mínima energia é a melhor solução para o problema.

Com o objetivo de escapar de ótimos locais o *Simulated Annealing* permite movimentos que resultam em soluções de pior qualidade que a solução atual. A probabilidade de realizar tais movimentos é reduzida durante a busca.

O método *Simulated Annealing* inicializa sua busca a partir de uma solução inicial qualquer e de uma temperatura inicial (T_0) definida a priori. A cada temperatura do método é aplicado o algoritmo de Metropolis (Metropolis *et al.*, 1953) até que seja atingido o equilíbrio térmico nesta temperatura, que é obtido após um número máximo de iterações (SA_{max}), definido a priori.

A cada iteração, uma solução vizinha da solução corrente é gerada aleatoriamente. Se essa solução for melhor que a solução corrente, continua-se a busca a partir do vizinho gerado. Caso contrário, existe uma probabilidade de aceitar um movimento que apresenta uma piora da solução e, que está associada ao valor da temperatura corrente. A probabilidade de aceitar uma solução que seja pior que a corrente é dada por $e^{-\Delta/T}$, sendo Δ a variação do valor da função objetivo ao mover-se da solução corrente para a solução vizinha, e T é a temperatura corrente. Para valores altos de temperatura há uma

grande probabilidade de se aceitar uma solução de pior custo, sendo que para temperaturas baixas, esta probabilidade é praticamente nula Chaves (2009).

A temperatura é diminuída de forma gradativa por uma taxa de resfriamento α após SA_{max} iterações, por exemplo, $T_k = \alpha \times T_{k-1}$ e $\alpha \in [0; 1]$. Isto faz com que o método se comporte de forma aleatória no início do processo de busca e à medida que a temperatura se aproxima de zero tende a um método de refinamento, pois a probabilidade de aceitação de movimentos de piora diminui, significativamente. Assim sendo, o critério de resfriamento é o responsável por gerar o equilíbrio entre diversificação e intensificação no *Simulated Annealing*.

A principal fase do *Simulated Annealing* é a intermediária. Por isso, uma técnica muito utilizada é o reaquecimento da temperatura, permitindo que o processo de busca seja executado por mais algumas iterações antes que a temperatura se aproxime de zero, por exemplo, pode-se aumentar a temperatura a 30% da temperatura inicial, quando esta atingir o valor de 0,1 pela primeira vez (Chaves, 2009).

Existem duas formas de se estabelecer o critério de parada para o método *Simulated Annealing*. A primeira é quando a temperatura chega a um valor próximo de zero ($T = 0; 001$) e assim, nenhuma solução de piora é mais aceita, evidenciando o encontro de um ótimo local. Na segunda pode ser fixado um número máximo de iterações.

3.4 Cluster Search e Simulated Annealing aplicados ao DARP

Nesse trabalho é aplicado o algoritmo *Cluster Search* – CS desenvolvido por Oliveira (2004) em conjunto com *Simulated Annealing* desenvolvido por Kirkpatrick *et al.* (1983). Para que a aplicação destes métodos fosse possível, utilizou-se ainda, métodos para realizar a distribuição e programação das rotas e um método chamado de estrutura de vizinhança para permitir a geração da solução vizinha, necessária para o funcionamento adequado do *Simulated Annealing*.

A busca por modelos e métodos capazes de resolver o *DARP*, tem sido uma constante entre os pesquisadores que estudam este tipo de problema. Por isso, a aplicação dos métodos escolhidos possibilita encontrar novas soluções que serão avaliadas por meio de comparação, com os melhores resultados encontradas na literatura para as instâncias utilizadas.

3.4.1 Representação do *DARP*

Algumas adaptações na forma de representar o problema são necessárias, para que o algoritmo possa efetuar a roteirização e a programação inicial e, a partir desta representação, resolver o problema em questão, ou seja, precisa ser definida uma forma estrutural para o problema.

Inicialmente, é criado um vetor com os dados de todos os locais descritos nas instâncias (pontos de coleta, entrega, e garagens). A garagem é colocada no início e no final desse vetor. Cada posição desse vetor é preenchida com uma estrutura do tipo *Location* (Figura 3.3), que são armazenadas as coordenadas X e Y do ponto, as janelas de tempo, etc. A seguir, é apresentado um vetor com $2n$ pontos (sendo n o número de clientes) e as garagens de origem e destino.

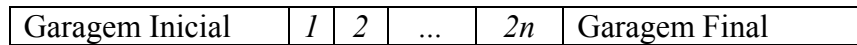


Figura 3.3 – Estrutura dos pontos

A partir de então, são formadas as rotas (Figura 3.4), que são representadas por uma lista de números inteiros fazendo referência às *Locations* (Figura 3.3). Essa lista apresenta a sequência de atendimento na rota. Obviamente, as garagens não entram nessa lista, pois são implícitas no início e final de toda rota. Cada rota ainda armazena algumas informações básicas, como distância percorrida, tempo de duração, etc. A rota mostrada na Figura 3.4, diz que o veículo partirá da garagem, passará pelos pontos 10, 34, 5 e 29, nessa ordem, e retornará a garagem.

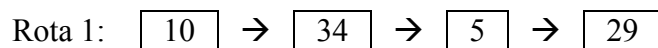


Figura 3.4 – Rota de atendimento

Por fim, as soluções são representadas por uma lista de rotas (Figura 3.5). Obviamente, cada solução (*Solution*) também armazena algumas informações extras, como custo total, distância total percorrida, número de veículos utilizados, número de clientes atendidos, etc. Logo, considerando a existência de m veículos, uma solução pode ser representada como descrito abaixo.

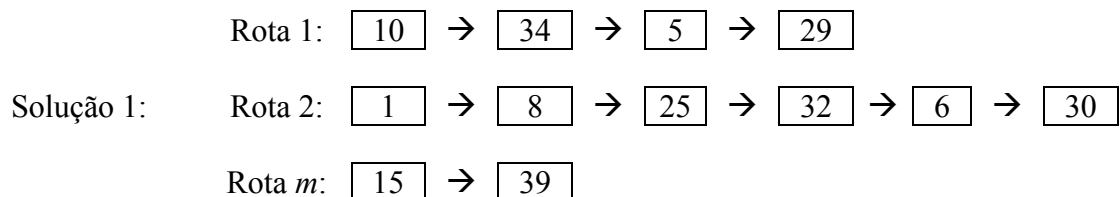


Figura 3.5 – Lista de rotas

3.4.2 Detalhamento da Implementação do CS-SA

Primeiramente, para verificar a similaridade entre duas soluções, foi utilizada a distância de *Hamming* (Hamming, 1950), definida como sendo o número de arcos diferentes entre as soluções. Considerando a Solução 1 apresentada na Figura 3.5 (e o ponto 0 como sendo a garagem), pode-se definir os seguintes arcos:

Solução 1: (0,10), (10,34), (34,5), (5,29), (29,0), (0,1), (1,8), (8,25), (25,32), (32,6), (6,30), (30,0), (0,15), (15,39), (39,0)

Considerando agora uma Solução 2, com os seguintes arcos:

Solução 2: (0,10), (10,5), (5,29), (29,34), (34,0), (0,1), (1,8), (8,25), (25,32), (32,6), (6,30), (30,0), (0,15), (15,39), (39,0)

A distância entre as soluções 1 e 2 é: $d_{Ham}(S_1, S_2) = 3$ (arcos em azul)

Seguindo o fluxograma do *Cluster Search* (Figura 3.1), são criados então os *clusters* iniciais. Assim, para cada *cluster* é criada uma solução inicial, de forma que todas as requisições de transportes sejam distribuídas entre os veículos disponíveis. É realizada ainda a programação dos atendimentos, ou seja, nesta fase é programada a ordem que os usuários serão atendidos.

A heurística de distribuição consiste em criar m rotas vazias atribuindo para cada rota um veículo a fim de que este efetue certo número de requisições desde que não exceda a sua respectiva capacidade. O passo seguinte do método realiza a alocação de todas as requisições de transportes dos clientes (pontos de embarque e seus respectivos pontos de desembarque). Essa distribuição dos atendimentos nas m rotas criadas no passo anterior é feita aleatoriamente, porém de maneira uniforme, ou seja, as n requisições são divididas igualmente entre os m veículos. Entretanto, podem ocorrer casos onde a divisão de n por m não é inteira, nesses casos, a última rota deve atender um número superior de requisições. A seleção das requisições, as rotas que as atenderão e a posição de inserção das requisições nas rotas é realizada de forma aleatória, vale ressaltar que essas inserções respeitam a restrição de precedência implícita na requisição de transporte do cliente em questão, ou seja, um local de embarque sempre deverá ser inserido antes de seu respectivo local de desembarque. As rotas começam e terminam em garagens específicas, desta forma, é possível afirmar que os locais referentes a estas, sempre serão alocados como primeiro e último pontos em cada uma das rotas criadas (Mauri e Lorena, 2009). A Figura 3.6 apresenta um pseudocódigo desta heurística.

1. CRIAR (m rotas vazias, e atribuí-las aos m veículos);
2. CRIAR (uma lista L com todas as requisições de transporte dos clientes);
3. PARA (cada rota k, k = 1,2,...,m) FAÇA
4. SELECIONAR ($\lfloor n/m \rfloor$ requisições de L);
5. PARA (cada requisição selecionada) FAÇA
6. Pos1 ← posição qualquer da rota k;
7. Pos2 ← posição qualquer da rota k, porém posterior a Pos1;
8. INSERIR (o ponto de embarque em Pos1);
9. INSERIR (o ponto de desembarque em Pos2);
10. FIM-PARA;
11. INSERIR (a garagem de origem do veículo k no início da rota k);
12. INSERIR (a garagem de destino do veículo k no fim da rota k);
13. REMOVER (as requisições selecionadas de L);
14. FIM-PARA;

Fonte: Mauri e Lorena (2009)

Figura 3.6 – Heurística de distribuição

Na Figura 3.7 é apresentado o outro método necessário para a construção da solução inicial. Este método é uma heurística de programação, derivada da apresentada por Cordeau e Laporte (2003b) e, consiste em definir a ordem pela qual se executa o atendimento das requisições de transportes, ou seja, determinar os horários de chegada e partida nos locais de atendimento, entre outros. Ao realizar a programação dos atendimentos este método procura ainda reduzir as violações nas janelas de tempo, a duração das rotas, e os tempos de viagem dos clientes.

1. $B_0 \leftarrow e_0; D_0 \leftarrow B_0;$
2. CALCULAR (A_i, B_i, W_i, D_i, Q_i para cada ponto $v_i \in V_k$ e $v_i \neq v_0$);
3. CALCULAR (F_0);
4. $B_0 \leftarrow e_0 + \min \left\{ F_0, \sum_{0 < p \leq z} W_p \right\}; D_0 \leftarrow B_0;$
5. ATUALIZAR (A_i, B_i, W_i, D_i para cada ponto $v_i \in V_k$ e $v_i \neq v_0$);
6. CALCULAR (R_i para cada ponto $v_i \in V_k$ e $v_i \in P$);
7. PARA (cada ponto $v_i \in V_k$ e $v_i \in P$) FAÇA
8. CALCULAR (F_i);
9. $B_i \leftarrow B_i + \min \left\{ F_i, \sum_{i < p \leq z} W_p \right\};$
10. $D_i \leftarrow B_i + s_i; W_i \leftarrow B_i - A_i;$
11. ATUALIZAR (A_j, B_j, W_j, D_j para cada ponto $v_j \in V_k$ e v_j posterior a v_i);
12. ATUALIZAR (R_j para cada ponto $v_j \in V_k, v_j \in P$ e v_{n+j} posterior a v_i);
13. FIM-PARA;

Fonte: Mauri e Lorena (2009)

Figura 3.7 – Heurística de programação

A heurística de programação resume-se em alguns passos que servem para efetuar a programação (sequência) de atendimentos dos clientes. Inicialmente, define-se o horário de partida da garagem de origem (G^-) que é marcado para o horário de início da respectiva janela de tempo. Após essa definição, os demais cálculos (horários de chegada, início do serviço, tempo de espera, horário de partida e carga do veículo) são realizados para todos os pontos i ($\forall_i \in K$) seguintes na rota. Feito isso, o método calcula o atraso para a partida da garagem de origem (G^-), e então o horário de partida é ajustado de forma a não aumentar as violações nas janelas de tempo. A partir de então, é realizada uma atualização nos horários para todos os pontos posteriores à garagem de origem (G^-), além disso, é efetuado o cálculo do tempo de viagem dos clientes. Finalmente, para cada ponto de embarque i ($\forall_i \in P$) pertencente à rota em análise, é calculado o seu respectivo atraso (sendo o horário de início do serviço ajustado de forma a reduzir a duração da rota e o tempo de viagem dos clientes, e não aumentar as violações nas janelas de tempo), e os horários de todos os pontos posteriores são atualizados, bem como o tempo de viagem dos clientes cujo desembarque é posterior ao ponto de embarque em questão (Mauri e Lorena, 2009).

A partir de então, o *Simulated Annealing*, baseado no proposto por Mauri e Lorena (2009), é executado, e a cada temperatura, a solução corrente (não a melhor) é enviada ao *Cluster Search*. A Figura 3.8 apresenta um pseudocódigo do *Simulated Annealing* implementado neste trabalho. Pode-se notar que o *Cluster Search* é chamado na linha 20 desse algoritmo, ou seja, a cada temperatura.

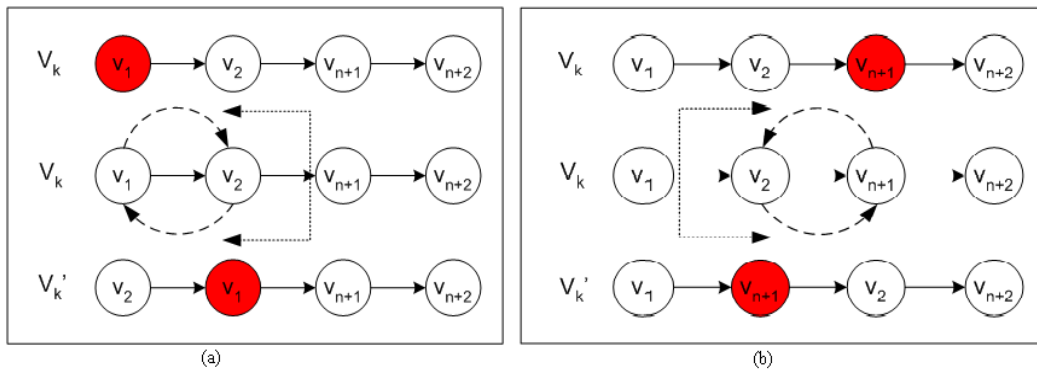
1. DADO (α , S_{Amax} , T_0 e T_c) FAÇA
2. GERAR (uma solução S através da heurística de distribuição);
3. APLICAR (a heurística de programação em todas as rotas de S);
4. $S^* \leftarrow S$; {Melhor solução obtida até então}
5. $\text{IterT} \leftarrow 0$; {Número de iterações na temperatura T }
6. $T \leftarrow T_0$; {Temperatura corrente}
7. ENQUANTO ($T > T_c$) FAÇA
8. ENQUANTO ($\text{IterT} < S_{\text{Amax}}$) FAÇA
9. $\text{IterT} \leftarrow \text{IterT} + 1$;
10. GERAR (um vizinho qualquer S' através de um dos mov. de troca);
11. APLICAR (a heurística de programação em todas as rotas de S');
12. $\Delta \leftarrow f(S') - f(S)$;
13. SE ($\Delta < 0$) $S \leftarrow S'$;
14. SE ($f(S') < f(S^*)$) $S^* \leftarrow S'$; FIM-SE
15. SENÃO
16. TOMAR ($x \in [0,1]$);
17. SE ($x < e^{-\Delta/T}$) $S \leftarrow S'$; FIM-SE
18. FIM-SE
19. FIM-ENQUANTO
20. **EXECUTAR-CS (solução corrente S);**
21. $T \leftarrow \alpha * T$; $\text{IterT} \leftarrow 0$;
22. FIM-ENQUANTO
23. $S \leftarrow S^*$;
24. RETORNAR (S);

Fonte: Adaptado de Mauri e Lorena (2009)

Figura 3.8 – Algoritmo Simulated Annealing utilizado no CS

Como estrutura de vizinhança no *Simulated Annealing* (linha 10, Figura 3.8) foram utilizados três métodos distintos que efetuam os movimentos de troca, necessários para que o método possa modificar a solução corrente a cada iteração, buscando com isso, melhorias na função objetivo de forma que o método não fique “preso” a ótimos locais (Mauri e Lorena, 2009). Esses métodos são:

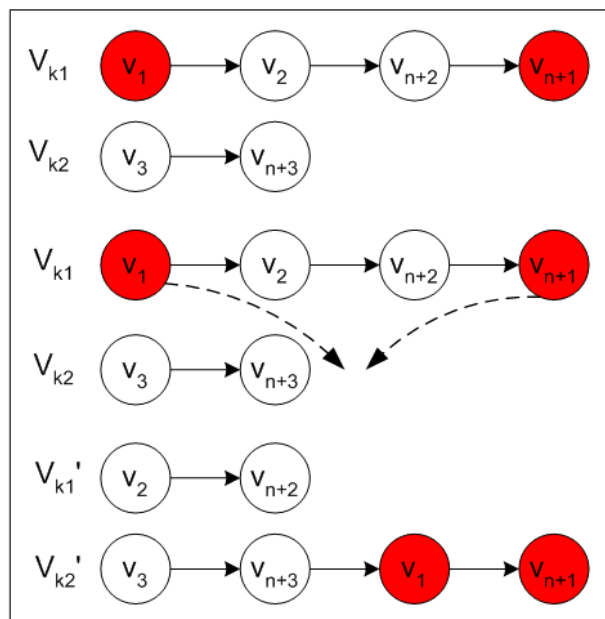
- *Movimento Re-ordenar rota*: Consiste em trocar a ordem de um ponto de uma rota, exceto os pontos que representam as garagens de início e fim da rota. Este ponto é selecionado aleatoriamente e retirado da posição em que se encontra e reinserido em outra posição desta mesma rota. A Figura 3.9 ilustra esse movimento. Como apenas um ponto é selecionado deve existir sempre o cuidado para que caso este seja um ponto de embarque, o método deve garantir que a reinserção deste seja antes do respectivo ponto de desembarque, como pode ser visto na Figura 3.9a. Já se o escolhido for de desembarque o método deve permitir que este seja reinserido após o ponto de embarque correspondente (ver Figura 3.9b).



Fonte: Mauri (2008)

Figura 3.9 – Movimento reordenar rota

- Movimento Re-alocar ponto:* O passo inicial consiste em selecionar duas rotas quaisquer que compõem a solução. No passo seguinte, uma requisição qualquer (embarque e desembarque) é retirada de uma rota e, reinserida em outra em uma posição qualquer. A extração dos pontos na rota de origem é feita de forma simultânea, porém a alocação destes pode ser feito de forma individual, respeitando sempre a restrição de precedência, como mostra a Figura 3.10.

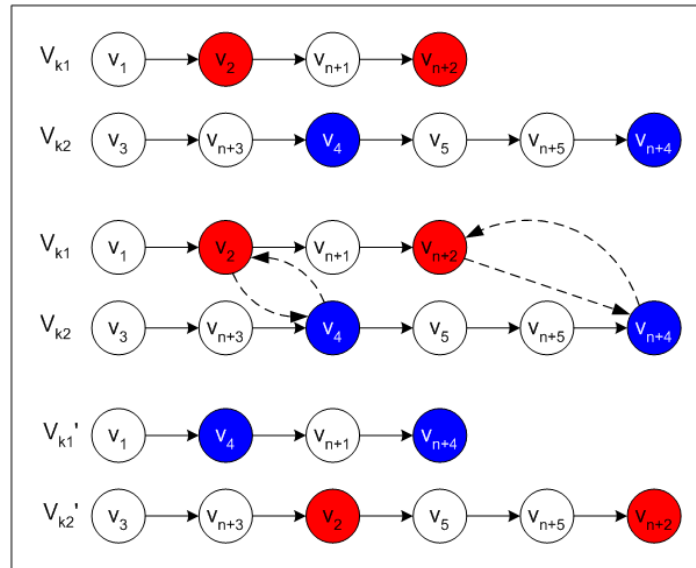


Fonte: Mauri (2008)

Figura 3.10 – Movimento realocar pontos

- Movimento Trocar pontos:* Consiste basicamente em selecionar duas rotas quaisquer pertencentes à solução, escolher uma requisição de transporte (ponto de embarque e seu respectivo ponto de desembarque) qualquer em cada uma das duas rotas, e trocá-las. Como essa troca é feita de forma simultânea, pode-se

garantir que, o ponto de embarque i será sempre anterior ao seu respectivo ponto de desembarque $n+i$ (Mauri e Lorena, 2009). Ver Figura 3.11.



Fonte: Mauri (2008)

Figura 3.11 – Movimento trocar pontos

Cada solução vizinha no *Simulated Annealing* (linha 10, Figura 3.8) é gerada por apenas um desses movimentos, sendo a sua escolha feita de forma aleatória, porém uniformemente distribuída, possibilitando assim uma boa diversidade entre as soluções intermediárias geradas, e conseqüentemente uma boa exploração do espaço de soluções (Mauri, 2008).

Após as definições estruturais que garantem a geração das soluções iniciais, estas prontas para serem analisadas e, possivelmente, modificadas pelos métodos principais, que neste caso é o *Simulated Annealing* e o *Cluster Search*.

O *Cluster Search* (CS) foi implementado de acordo com o apresentado em Chaves (2009). Vale destacar que antes mesmo da execução do *Simulated Annealing* (SA), as soluções “centro” de *clusters* já foram criadas (como descrito anteriormente). Ao final da execução do SA, ou seja, do método CS-SA como um todo a melhor solução centro de *cluster* é tomada como solução final para o problema. O algoritmo EXECUTAR-CS é apresentado na Figura 3.12.

1. DADA (uma solução S);
2. ENCONTRAR (o cluster C_j mais similar a S);
3. $v_j \leftarrow v_j + 1$;
4. ATUALIZAR (o centro do cluster C_j);
5. SE $v_j \geq \lambda$ ENTÃO
6. $v_j \leftarrow 1$;
7. SE $r_j \geq r_{\max}$ ENTÃO
8. APLICAR (perturbação em c_j)
9. $r_j \leftarrow 0$;
10. SENÃO
11. APLICAR (busca local – encontrar c_j');
12. SE $f(c_j') < f(c_j)$ ENTÃO
13. $c_j \leftarrow c_j'$;
14. $r_j \leftarrow 0$;
15. SENÃO
16. $r_j \leftarrow r_j + 1$;
17. FIM-SE;
18. FIM-SE;
19. FIM-SE;

Figura 3.12 – Algoritmo EXECUTAR-CS

Como mencionado anteriormente, a determinação do *cluster* mais similar (linha 2, Figura 3.12) é dada pela menor distância de *Hamming* (Hamming, 1950). A atualização do centro do *cluster* (linha 4 Figura 3.12) é dada pelo algoritmo apresentado na Figura 3.13. A ideia desse algoritmo é bastante simples e consiste em pegar a melhor solução entre o centro c_j e a nova solução S , e a partir desta, selecionar um ponto de coleta qualquer (e seu respectivo ponto de entrega) em uma rota qualquer, e procurar pela melhor posição para esses pontos nas demais rotas.

1. SE o custo de S for menor que o custo de c_j ENTÃO
2. $c_j \leftarrow S$;
3. FIM-SE;
4. SELECIONAR (uma rota i qualquer de c_j);
5. SELECIONAR (um ponto p qualquer e o respectivo $n+p$ na rota i);
6. REMOVER (os pontos p e $n+p$ da rota i);
7. PARA (todas as rotas $r \neq i$) FAÇA
8. INSERIR (o ponto p em todas as posições da rota r);
9. INSERIR (o ponto $n+p$ em todas as posições da rota r posteriores a p e que não violem o tempo de viagem do cliente p);
10. ARMAZENAR (as posições que resultam na melhor solução);
11. FIM-PARA;
12. INSERIR (os pontos p e $n+p$ nas melhores posições armazenadas);

Figura 3.13 – Algoritmo utilizado na atualização dos centros de clusters

A perturbação apresentada na linha 8 do *EXECUTAR-CS* (Figura 3.12) é dada por uma simples aplicação do “Movimento trocar pontos” (Figura 3.9). Por fim, a busca local

(linha 11 do EXECUTAR-CS, Figura 3.12) utilizada para intensificar a busca em *clusters* promissores é apresentada na Figura 3.14. É interessante destacar ainda que, a busca local é aplicada por rota, evitando assim um alto tempo de processamento.

1. PARA (cada rota pertencente a c_j) FAÇA
2. ENQUANTO (melhorar a solução) FAÇA
3. PARA (todos os pontos de c_j) FAÇA
4. INSERIR (o ponto p em todas as posições da rota);
5. INSERIR (o ponto $n+p$ em todas as posições posteriores a p e que não violem o tempo de viagem do cliente p);
6. ARMAZENAR (os pontos e as posições que resultam na melhor solução);
7. FIM-PARA;
8. INSERIR (os pontos nas melhores posições armazenadas);
9. FIM-ENQUANTO;
10. FIM-PARA;

Figura 3.14 – Busca local

3.5 Considerações Finais

A formulação matemática apresentada neste capítulo é uma adaptação feita por Mauri e Lorena (2009) a partir de outras formulações encontradas na literatura, como por exemplo, em Cordeau e Laporte (2003b), Bergvinsdottir *et al.* (2004), Cordeau (2006). Mauri e Lorena (2009) formularam o *DARP* de maneira “relaxada”, tratando os requisitos essenciais como parte da função a ser minimizada, e não como restrições do modelo. Se por um lado essa abordagem torna a resolução do problema menos árdua, do outro, permite que soluções inválidas sejam obtidas. A obtenção de soluções inválidas pode ser interessante em alguns casos e também justificada devido à limitação dos modelos exatos em resolver, de maneira satisfatória, um problema de dimensões reais, assim, uma solução com violações mínimas nos requisitos essenciais pode ser aceitável (Cordeau, 2003b).

Neste Capítulo é possível observar que a construção de modelos matemáticos é resultado de uma busca exaustiva, a fim de se conhecer todos os aspectos que caracterizam o *DARP*, bem como o seu grau de relevância. Observa-se ainda, que a procura pelo equilíbrio entre os custos operacionais e as inconveniências dos clientes é uma equação difícil de ser resolvida, sobretudo, quando o nível de serviço é tão importante como é o caso do *DARP*.

4 Avaliação dos Experimentos Computacionais

Este capítulo descreve os resultados dos experimentos computacionais realizados, a fim de validar a aplicação do CS-SA na resolução do *DARP*. Os resultados foram comparados com os de maior relevância, encontrados na literatura pesquisada, que utilizaram as mesmas instâncias objeto desta pesquisa.

Para testar o CS-SA foram realizados vários experimentos, sendo utilizadas as instâncias apresentadas por Cordeau e Laporte (2003b), disponíveis em: <http://www.hec.ca/chairedistributique/data/darp/>, que são referências em inúmeros trabalhos de considerável importância para resolução do *DARP* (Bergvinsdottir, 2004; Jorgensen *et al.*, 2007; Mauri e Lorena, 2009). Estão disponíveis 20 instâncias, combinadas entre 24 a 144 requisições de transporte (48 a 288 pontos) e 3 a 13 veículos, sendo as primeiras instâncias (R1a – R10a) formadas por janelas de tempo “largas” e o restante (R1b – R10b), por janelas de tempo mais “estreitas”.

A estrutura dessas instâncias é apresentada na Figura 4.1. Cada requisição de transporte é representada por dois pontos, um de embarque e outro de desembarque, ou seja, uma requisição é formada, por exemplo, pelos dados da linha 76 que representa o ponto de embarque e pela linha 152 que representa o respectivo ponto de desembarque. Os dados da primeira linha informam as características gerais da instância:

- A:** número de veículos disponíveis;
- B:** número de pontos (coleta e entrega) a serem atendidos, ou seja, $B/2$ = número de clientes;
- C:** tempo máximo de duração de uma rota (minutos);
- D:** capacidade dos veículos (número de assentos); e,
- E:** tempo máximo de viagem (tempo máximo que um cliente pode ficar dentro do veículo).

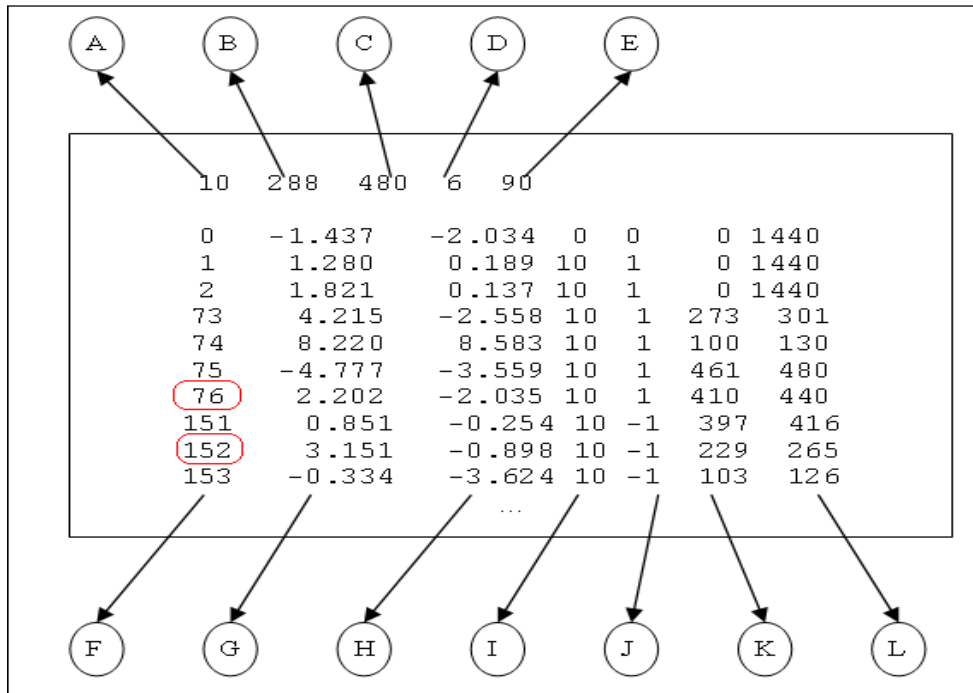


Figura 4.1 – Estrutura das instâncias utilizadas

Para cada ponto de embarque ou desembarque têm-se as seguintes informações:

F: identificador do ponto. Sendo, 0 a garagem; de 1 até $B/2$ os pontos de coleta, e de $((B/2)+1)$ até B os respectivos pontos de entrega;

G: coordenada X do ponto;

H: coordenada Y do ponto;

I: tempo de serviço no ponto;

J: informa se o ponto refere-se a garagem quando igual a 0 (zero), embarque ou ponto de coleta quando igual a 1 (um) e, desembarque ou ponto de entrega quando igual a (-1) (menos um);

K: horário de início da janela de tempo (minutos); e,

L: horário de término da janela de tempo (minutos).

Como exemplo, o ponto 76 corresponde a um embarque com carga q_i igual a 1 (um) e o início do serviço (janela de tempo) deverá ocorrer no minuto 410, ou seja, às 6 (seis) horas e 50 minutos e o término da janela de tempo para o respectivo embarque as 7 horas e 20 minutos. O ponto 152 representa o desembarque correspondente ao ponto de embarque 76.

Essas instâncias representam problemas com garagem única e frota homogênea, e não adotam o conceito de tempo máximo de espera. Os parâmetros utilizados pelo CS-SA, em todos os experimentos foram $T_0 = 20000$, $\alpha = 0.975$, $T_c = 0.01$, $SA_{max} = 1000$, $\lambda = 7$, $r_{max} = 3$, e o número de *clusters* = 10.

Já a escolha dos valores das penalizações aplicadas na função objetivo, foi baseada em uma análise apresentada por Jorgensen *et al.* (2007), porém, a penalização para as violações foi mais “pesada” (assim como em Mauri e Lorena, 2009), visando evitar a obtenção de soluções inválidas para o problema. Os pesos utilizados foram: $\omega = [8, 0, 1, 3, 1]$ e $\rho = [1500, 1500, 1500, 1500, 1500]$. Todos os experimentos foram realizados em um PC com processador AMD Athlon 64 3500 de 2.2 GHz e 1GB de memória RAM. Toda a implementação foi desenvolvida na linguagem C++.

As soluções obtidas por Cordeau e Laporte (2003b) foram executadas em um PC Intel Pentium 4 de 2.0 GHz; as obtidas por Jorgensen *et al.* (2007) em um PC Celeron de 2.0 GHz; e as obtidas por Mauri e Lorena (2009) em um laptop Toshiba A10 S127 com processador Intel Celeron de 2.0 GHz e 256MB de memória RAM.

4.1 Descrição dos Resultados

Foram realizados 5 (cinco) testes para cada instância. A Tabela 4.1 apresenta uma comparação direta dos resultados obtidos com os apresentados por Mauri e Lorena (2009). Nessa tabela, a coluna *Melhor $f(S)$* indica o valor da função objetivo da melhor solução encontrada nos testes realizados para cada instância. A coluna *$f(S)$ média* apresenta a média aritmética dos 5 (cinco) testes, e a coluna *Desvio* é obtida pela Equação 4.1. Em todos esses testes o número de veículos utilizados foi igual ao disponível, todas as requisições de transportes foram atendidas, e o mais importante, todas as soluções obtidas foram válidas, ou seja, todas as restrições foram satisfeitas.

$$\text{Desvio} = \left(\frac{f(S) \text{ média} - \text{Melhor } f(S)}{\text{Melhor } f(S)} \right) * 100 \quad (4.1)$$

Analisando a Tabela 4.1, percebe-se que, de uma forma geral, o CS-SA apresentou os melhores resultados. Sendo que apenas para as instâncias *R9a* e *R10b* as soluções, assim como as médias do CS-SA, foram piores que as apresentadas por Mauri e Lorena (2009), porém o desvio padrão das soluções obtidas com o CS-SA foi bem inferior apresentados pelo método SA. Desta forma, é importante afirmar que mesmo não apresentando soluções melhores para essas instâncias o CS-SA

Já em termos de robustez, o SA e o CS-SA apresentam resultados bastante similares, pois apresentaram desvios médios de 1,86% e 1,16%, respectivamente.

Observa-se ainda na Tabela 4.1 que o Desvio calculado pela equação 4.1 apresenta valores regulares em todas as instâncias para o método CS-SA. Já no método SA esses valores variam de 0,36% à 10,96%. Isto significa dizer que as soluções do CS-SA apresentam menor variação entre execuções diferentes.

Tabela 4.1 – Comparação do CS-SA com o SA proposto por Mauri e Lorena (2009)

Instância	SA			CS-SA		
	$f(S)$ média	Melhor $f(S)$	Desvio (%)	$f(S)$ média	Melhor $f(S)$	Desvio (%)
R1a	3.721,57	3.677,91	1,19	3.684,58*	3.677,91*	0,18*
R2a	7.101,87	70.17,34	1,20	7.070,80*	7.000,97*	1,00*
R3a	11.982,18	11.873,76	0,91*	11.923,29*	11.787,38*	1,15
R4a	13.982,52	13.725,92	1,87	13.738,45*	13.530,93*	1,53*
R5a	16.006,82	15.736,66	1,72	15.881,79*	15.654,90*	1,45*
R6a	20.607,77	20.465,39	0,70*	20.330,39*	20.075,38*	1,27
R7a	5.718,46	5.610,05	1,93	5.596,31*	5.523,08*	1,33*
R8a	11.554,71	11.343,19	1,86*	11.339,26*	11.129,60*	1,88
R9a	17.345,60*	15.632,09*	10,96	17.936,26	17.347,66	3,39*
R10a	23.104,61	22.430,00	3,01	22.466,20*	22.024,21*	2,01*
R1b	3.407,87	3.379,74	0,83*	3.389,97*	3.343,39*	1,39
R2b	5.925,35	5.889,56	0,61	5.840,77*	5.808,80*	0,55*
R3b	11.045,26	11.006,12	0,36*	10.877,43*	10.837,09*	0,37
R4b	12.856,75	12.807,87	0,38	12.686,25*	12.650,47*	0,28*
R5b	14.874,48	14.544,13	2,27	14.528,47*	14.397,45*	0,91*
R6b	18.795,79	18.518,82	1,50	18.632,62*	18.408,39*	1,22*
R7b	5.202,93	5.136,37	1,30	5.170,64*	5.132,52*	0,74*
R8b	10.791,67	10.703,17	0,83	10.548,51*	10.469,19*	0,76*
R9b	15.180,91	15.013,71	1,11	14.923,10*	14.852,27*	0,48*
R10b	20.492,56*	19.969,15*	2,62	20.496,60	20.240,19	1,27*

* Melhores resultados.

A Tabela 4.2 apresenta as melhores soluções obtidas pelo CS-SA para os testes realizados. A boa adaptação deste algoritmo na resolução do *DARP* fica evidente quando se compara as soluções encontradas através da aplicação de métodos utilizados frequentemente, como é o caso do *Simulated Annealing* (Mauri e Lorena, 2009), Busca Tabu (Cordeau e Laporte, 2003b) e Algoritmo Genético (Jorgensen *et al.*, 2007).

Tabela 4.2 – Melhores soluções obtidas pelo CS-SA

Instância	Distância percorrida	Duração das rotas (min)	Tempo de espera (min)		Tempo de viagem (min)		CPU (min)
			Total	Médio	Total	Médio	
R1a	252,79	831,30	98,51	2,05	241,93	10,08	0,56
R2a	431,26	1.957,29	566,03	5,90	342,52	7,14	0,79
R3a	845,31	2.474,62	189,30	1,31	786,98	10,93	1,07
R5a	1.078,18	3.875,25	397,06	1,65	919,04	7,66	1,54
R9a	1.029,96	3.241,21	51,25	0,24	1.938,51	17,95	1,98
R10a	1.440,75	4.487,98	167,22	0,58	1.947,66	13,53	2,63
R1b	239,27	731,95	12,68	0,26	228,19	9,51	0,56
R2b	436,29	1.422,84	26,54	0,28	289,69	6,04	0,92
R5b	1.050,25	3.479,19	28,94	0,12	829,09	6,91	1,74
R6b	1.306,93	4.276,82	89,89	0,31	1.195,41	8,30	1,81
R7b	386,38	1.133,25	26,87	0,37	293,78	8,16	0,78
R9b	1.048,42	3.261,30	52,88	0,24	1.050,23	9,72	1,90
R10b	1.382,19	4.281,61	19,42	0,07	1.627,22	11,30	2,59
TOTAL	10.927,98	35.454,61	1.726,59	13,38	11.690,25	127,23	18,87

As avaliações para comparação estão agrupadas de acordo com a função objetivo, definida no modelo apresentado na Seção 3.3, para isso, foram avaliados aspectos como distância total percorrida em cada para atendimento da demanda em cada instância, tempo de duração da rota, tempo médio de viagem e tempo de processamento gasto para o método encontrar a melhor solução.

Comparando a distância total percorrida encontrada pelo CS-SA com a obtida com a utilização do método *Simulated Annealing* (SA) apresentado por Mauri e Lorena (2009), nota-se que o CS-SA apresentou um aumento de 0,41% na distância total percorrida pelos veículos. Quando a comparação é entre o CS-SA e os resultados obtidos pelo método da Busca Tabu (Cordeau e Laporte, 2003b), os resultados do CS-SA apresentam aumento de 64,21% na distância total percorrida. Como as distâncias percorridas pelos veículos não são apresentadas em Jorgensen *et al.* (2007), não foi possível realizar nenhuma comparação. A Tabela 4.3 apresenta os resultados obtidos pelos métodos para cada instância.

Tabela 4.3 – Distância total percorrida

Instância	Distância percorrida		
	Busca Tabu (*)	Simulated Annealing (**)	CS-SA
R1a	190,02	252,79	252,79
R2a	302,08	437,45	431,26
R3a	532,08	831,74	845,31
R5a	636,97	1085,45	1.078,18
R9a	672,44	1064,23	1.029,96
R10a	878,76	1392,09	1.440,75
R1b	164,46	251,85	239,27
R2b	296,06	436,69	436,29
R5b	589,74	1.010,09	1.050,25
R6b	743,60	1.289,31	1.306,93
R7b	248,21	375,67	386,38
R9b	601,96	1.041,09	1.048,42
R10b	798,63	1.414,65	1.382,19
TOTAL	6.655,01	10.883,10	10.927,98

(*) – Cordeau e Laporte (2003b); (**) – Mauri e Lorena (2009)

No caso do *DARP*, o tempo de duração das rotas pode não ser proporcional à distância, ou seja, depende também do grau de influência que outros fatores como as janelas de tempo exercem sobre a mesma. Desta forma, mesmo a distância total percorrida obtida pelo CS-SA sendo maior que as demais, o tempo total de duração das rotas foi reduzido de forma significativa.

Como pode ser visto na Tabela 4.4, o tempo total de duração das rotas obtidos na aplicação do CS-SA foi reduzido quando comparados com os resultados encontrados com a aplicação do método da Busca Tabu, apresentado por Cordeau e Laporte (2003b), houve uma redução de 0,23%. Em relação às soluções obtidas por meio da aplicação do *Simulated Annealing* proposto por Mauri e Lorena (2009), a redução no tempo total de duração das rotas, foi 0,87%. Já em relação aos resultados alcançados pelo Algoritmo Genético apresentados por Jorgensen *et al.* (2007), o CS-SA foi capaz de reduzir em 12,48% no tempo de duração das rotas.

Tabela 4.4 – Tempo de duração das rotas

Instância	Duração das rotas (min)			
	CS-SA	Busca Tabu (*)	Simulated Annealing (**)	Algoritmo Genético (***)
R1a	831,30	881,16	831,3	1.039
R2a	1.957,29	1.985,94	1.992,34	1.994
R3a	2.474,62	2.579,35	2.404,67	2.781
R5a	3.875,25	3.869,95	3.920,25	4.274
R9a	3.241,21	3.155,49	3.258,66	3.526
R10a	4.487,98	4.480,1	4.475,42	5.025
R1b	731,95	965,06	738,42	928
R2b	1.422,84	1.564,74	1.428,44	1.710
R5b	3.479,19	3.595,63	3.654,02	4.336
R6b	4.276,82	4.072,47	4.318,33	5.227
R7b	1.133,25	1.097,25	1.095,67	1.316
R9b	3.261,30	3.249,29	3.315,28	3.676
R10b	4.281,61	4.040,99	4.332,69	4.678
TOTAL	35.454,61	35.537,42	35.765,49	40.510,00

(*) – Cordeau e Laporte (2003b); (**) – Mauri e Lorena (2009);

(***) – Jorgensen *et al.* (2007).

Analisando os aspectos relacionados à qualidade dos serviços prestados, de forma que as inconveniências aos clientes sejam minimizadas é possível notar que o método proposto apresenta resultados bastante promissores. Isto mostra reafirma as observações feitas por Mauri e Lorena (2009), que apresentaram o modelo como uma alternativa na busca por melhorias que garantam um melhor nível de serviço.

Na Tabela 4.5, encontram-se os resultados obtidos pelo CS-SA em relação ao tempo médio de viagem dos clientes. Quando comparados com os apresentados por Mauri e Lorena (2009), por meio da utilização do método Simulated Annealing, nota-se que o método CS-SA apresentou um aumento de 2,60% no tempo médio de viagem dos clientes. Em relação aos resultados do Algoritmo Genético, apresentados por Jorgensen *et al.* (2007), mostra uma redução de 74,69% nos mesmos e quando comparados com os resultados do método da Busca Tabu, apresentados por Cordeau e Laporte (2003b), ocorre uma redução de 79,95%.

Tabela 4.5 – Tempo médio de viagem

Instância	Tempo Médio de Viagem (min)			
	Simulated Annealing (**)	CS-SA	Algoritmo Genético (***)	Busca Tabu (*)
R1a	10,08	10,08	12,9	45,62
R2a	6,46	7,14	27,72	41,18
R3a	12,42	10,93	40,2	49,82
R5a	7,49	7,66	40,3	51,3
R9a	11,81	17,95	62,21	52,05
R10a	15,31	13,53	57,92	49,75
R1b	8,61	9,51	22,89	43,4
R2b	6,50	6,04	27,07	49,86
R5b	7,13	6,91	39,33	50,87
R6b	8,65	8,3	44,42	51,02
R7b	9,59	8,16	21,76	48,94
R9b	10,05	9,72	49,61	51,68
R10b	9,91	11,3	56,38	49,11
TOTAL	124,01	127,23	502,71	634,60

(*) – Cordeau e Laporte (2003b); (**) – Mauri e Lorena (2009);

(***) – Jorgensen *et al.* (2007).

O tempo médio de espera dos veículos pode ser visto como um fator gerador de inconveniências, tanto para o operador como para os clientes. Por exemplo, se o veículo atrasar a chegada para atendimento de uma requisição, o cliente é afetado de forma direta em sua programação, podendo gerar diversos transtornos ao mesmo. Do ponto de vista do operador pode significar tempo maior para execução dos serviços, implicando em aumento de custos, ou até mesmo sanções por parte do contratante, caso o operador seja um prestador do serviço. Quanto a esta característica, o CS-SA apresenta resultados expressivos, pois ao compará-lo com as soluções obtidas pelo método *Simulated Annealing* proposto por Mauri e Lorena (2009), o tempo médio de espera dos veículos foi reduzido em 7,85%. Ao fazer a mesma comparação com os resultados obtidos pelo método da Busca Tabu, apresentados por Cordeau e Laporte (2003b), esta abordagem conseguiu uma redução de 68,83%, no tempo médio de espera dos veículos. Já em relação aos resultados encontrados por Jorgensen *et al.* (2007) por meio do Algoritmo Genético, a redução é de 52,57%.

Esta redução significativa no tempo de espera dos veículos obtida pelo CS-SA mostra que, do ponto de vista da qualidade do serviço, a combinação destes métodos se apresenta como uma boa alternativa para resolução do *DARP*.

Tabela 4.6 – Tempo médio de espera

Instância	Tempo Médio de espera (min)			
	CS-SA	Simulated Annealing (**)	Algoritmo Genético (***)	Busca Tabu (*)
R1a	2,05	2,05	5,42	4,4
R2a	5,90	6,2	5,36	7,54
R3a	1,31	0,92	2,09	4,22
R5a	1,65	1,81	2,2	3,47
R9a	0,24	0,16	0,15	1,5
R10a	0,58	0,71	0,86	2,5
R1b	0,26	0,14	3,42	6,68
R2b	0,28	0,33	1,69	3,22
R5b	0,12	1,02	2,37	2,52
R6b	0,31	0,52	1,78	1,56
R7b	0,37	0	1,78	1,79
R9b	0,24	0,53	0,82	2,26
R10b	0,07	0,13	0,29	1,26
TOTAL	13,38	14,52	28,23	42,92

(*) – Cordeau e Laporte (2003b); (**) – Mauri e Lorena (2009);

(***) – Jorgensen *et al.* (2007).

Quanto ao tempo de processamento, o CS-SA apresenta ganhos expressivos, principalmente, quando observados os tempos gastos na tentativa de resolver o problema de forma exata, uma vez que o alto tempo gasto um dos principais aspectos que inviabiliza a resolução por métodos exatos. Por isso, os resultados obtidos neste trabalho mostram mais uma vez que, o CS-SA se apresenta como uma nova alternativa para resolver problemas de otimização de alta complexidade, como é o caso do *DARP*. Ao comparar os tempos de processamento utilizados para resolver as instâncias analisadas por meio do CS-SA, com os necessários para que o *Simulated Annealing* proposto por Mauri e Lorena (2009) resolva as mesmas e, percebe-se que houve uma redução de 16,65% nos mesmos. Em relação aos resultados derivados da utilização do Algoritmo Genético proposto por Jorgensen *et al.* (2007), a redução é de 96,14%. Já em relação aos resultados obtidos por Cordeau e Laporte (2003b), decorrentes da aplicação do algoritmo de Busca Tabu, ocorre uma redução de 96,21% no tempo de processamento, para resolver as mesmas instâncias. A Tabela 4.7 mostra os melhores tempos de processamento para as instâncias utilizadas. As avaliações feitas mostram ainda, que apesar do uso dos mesmos parâmetros utilizados pelo Simulated Annealing

proposto por Mauri e Lorena (2009), o CS-SA foi capaz de convergir para melhores resultados de forma mais rápida, para quase todos os casos.

Tabela 4.7 – Tempos de Processamento

Instância	Tempo de Processamento (min)			
	CS-SA	Simulated Annealing (**)	Algoritmo Genético (***)	Busca Tabu (*)
R1a	0,56	1	5,57	1,9
R2a	0,79	1,2	11,43	8,06
R3a	1,07	1,46	21,58	17,18
R9a	1,98	2,28	40,78	50,51
R5a	1,54	1,79	58,23	46,24
R10a	2,63	2,72	65,98	87,53
R1b	0,56	0,92	5,46	1,93
R7b	0,78	1,05	8,29	4,23
R2b	0,92	1,3	11,72	8,29
R9b	1,90	2,26	44,66	51,28
R5b	1,74	1,95	58,93	54,33
R6b	1,81	1,94	81,23	73,7
R10b	2,59	2,77	66,41	92,41
TOTAL	18,87	22,64	480,27	497,59

(*) – Cordeau e Laporte (2003b); (**) – Mauri e Lorena (2009);

(***) – Jorgensen *et al.* (2007).

A Figura 4.2 ilustra graficamente o comportamento do tempo de processamento dos métodos comparados para as instâncias *R1a* a *R10a*, onde as restrições de janela de tempo são mais “folgadas”. Como pode ser observado, os métodos CS-SA e o *Simulated Annealing* (SA) proposto por Mauri e Lorena (2009) apresentam pequenas variações no tempo de resolução para número diferente de requisições, ou seja, alterações no número de requisições não implicam em variações significativas no tempo de processamento utilizado. Já os métodos de Busca Tabu (BT) e Algoritmo Genético (AG) mostram uma característica bastante interessante, ou seja, o crescimento do tempo é uniforme quando o número de requisições é inferior a 72 requisições, mas para um número maior de requisições o tempo de processamento começa a variar de maneira distinta. Observa-se ainda que para instâncias com menor número de requisições, o método da Busca Tabu proposto por Cordeau e Laporte (2003b) resolve o problema com tempos inferiores aos do Algoritmo Genético (AG) proposto por Jorgensen *et al.* (2007), porém, para valores maiores este comportamento oscila de maneira diferente.

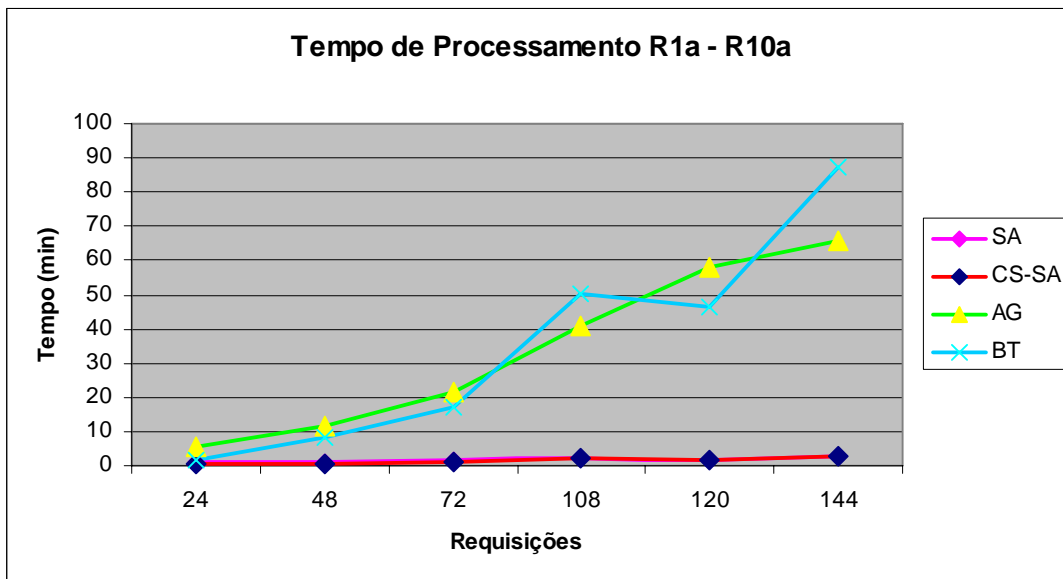


Figura 4.2 – Tempo processamento R1a a R10a

Como visto no início deste capítulo, as instâncias denominadas R1b a R10b possuem restrições de janela de tempo mais “estritas” ou “rígidas”. O gráfico da Figura 4.3 ilustra o comportamento dos métodos analisados na resolução dessas instâncias. Assim como no caso das instâncias com restrições mais “folgadas” o CS-SA e o SA se comportam de maneira uniforme para praticamente todos os casos, mesmo nas situações onde ocorre uma variação de outros parâmetros, além do número de requisições. Isto fica evidente nas instâncias *R9b* e *R10b*, em que, apesar do número de requisições ser igual, as janelas de tempo são diferentes e mesmo assim, não há grandes variações do tempo de processamento necessário para encontrar as soluções.

Já o método da Busca Tabu (BT) proposto por Cordeau e Laporte (2003b) e o Algoritmo Genético (AG) proposto por Jorgensen *et al.* (2007) apresentam distribuição semelhante apenas para os casos com menos de 48 requisições de transporte. Pode-se verificar, com isso, que o tempo de processamento não se dá em função apenas do número de requisições. Ao observar o gráfico na Figura 4.3, vê-se que para um mesmo número de requisições os métodos de Busca Tabu (Cordeau e Laporte, 2003b) e Algoritmo Genético (Jorgensen *et al.*, 2007) apresentam valores díspares. Isso ocorre devido a variações nas janelas de tempo (horário de início e fim dos serviços).

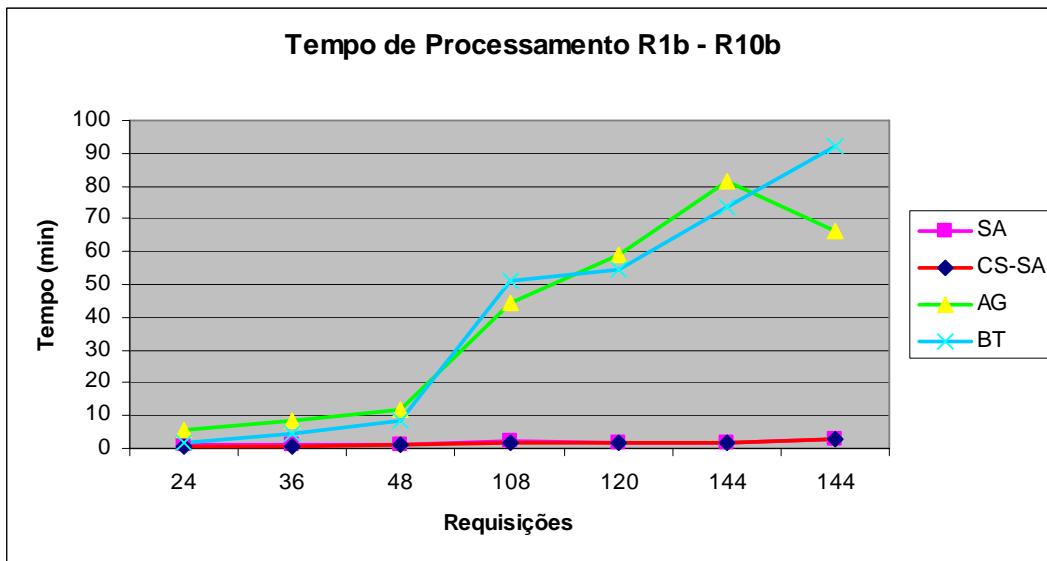


Figura 4.3 – Tempo processamento R1b a R10b

Comparando os gráficos das Figuras 4.2 e 4.3 percebe-se uma diferença comportamental significativa referentes aos tempos de processamento necessário para resolver as instâncias com janelas de tempo mais e menos rígidas.

4.2 Considerações Finais

Após analisar os resultados obtidos e efetuar as comparações necessárias, a fim de verificar o comportamento do CS-SA na resolução do *DARP*, é possível afirmar que, o método apresenta-se como uma ferramenta com potencial significativo para resolver problemas da classe *NP-difícil*, como é o caso dos problemas classificados como *dial-a-ride*. Algumas características ficam evidentes nas comparações feitas. Por exemplo, o CS-SA apresentou resultados melhores para quase todas as instâncias resolvidas quando comparado com o *Simulated Annealing* (SA) proposto por Mauri e Lorena (2009). Já nos casos em que o CS-SA não foi melhor, as soluções obtidas estiveram sempre muito próximas das encontradas pelo método *Simulated Annealing*. Isto pode ser justificado pelo fato de que em ambos os métodos, procurou-se resolver o mesmo modelo matemático.

Quando comparados os resultados obtidos pelo CS-SA com as soluções encontradas pelos métodos Busca Tabu (Cordeau e Laporte, 2003b) e Algoritmo Genético (Jorgensen *et al.*, 2007), observa-se uma melhoria considerável em relação ao nível de serviço e tempo de processamento utilizado. Assim é possível concluir que o CS-SA possui melhor desempenho quando o objetivo principal é a maximização da qualidade

dos serviços prestados ou a minimização da insatisfação dos clientes, por meio da redução das inconveniências.

A minimização dos custos fixos, tempo de viagem dos clientes, tempo médio de espera por atendimentos e da distância total percorrida, está relacionada com a escolha do método de resolução, juntamente com a forma de representação do *DARP*.

5 Conclusões e Recomendações

Os problemas de roteirização e programação de veículos, que tratam do transporte de pessoas portadoras de necessidades especiais de seus locais de origem aos seus respectivos destinos, são conhecidos genericamente na literatura como problemas *dial-a-ride (DARP)*. Ressalta-se que ao contrário dos serviços de táxis convencionais onde cada passageiro é atendido de maneira individual, no *DARP*, várias pessoas, com diferentes origens e destinos, podem ser transportadas ao mesmo tempo pelo mesmo veículo.

Por ser um serviço dedicado ao atendimento de pessoas portadoras de necessidades especiais, o nível de serviço torna-se um aspecto de fundamental importância. Essa característica é um fator que pode aumentar de forma significativa a complexidade dos modelos de resolução do *DARP*, inviabilizando a utilização de métodos exatos para resolver casos reais, em que, geralmente, o número de requisições de transporte é bastante expressivo.

Ao transportar pessoas, surge a necessidade de se garantir o melhor nível de serviço por meio da redução das inconveniências aos clientes. Desta forma, a garantia de um melhor nível de serviço, caminha de forma inversa ao objetivo principal de todo e qualquer operador do sistema, que tem como meta central o menor custo total.

Neste trabalho desenvolveu-se um procedimento para resolver o *DARP* por meio da utilização da metaheurística híbrida conhecido por *Cluster Search*. O emprego desses métodos tem sido justificado, principalmente, pela impossibilidade de se resolver tais problemas com a aplicação de métodos exatos, sobretudo quando envolve casos reais, limitando assim, o seu uso a problemas com aproximadamente 55 requisições de transporte, como visto em Dumas *et al.* (1991).

Como o objetivo principal do trabalho é avaliar a aplicabilidade de metaheurísticas para resolver problemas práticos, foi implementado então, métodos em linguagem computacional, a fim de verificar o comportamento e a adaptação dos algoritmos na resolução do *DARP*. A escolha desses métodos foi possível após um levantamento criterioso, que permitiu definir não somente, qual método deveria ser utilizado, mas também os modelos que apresentam maior similaridade com os problemas reais.

Como método de resolução, buscou-se implementar uma combinação de metaheurísticas, tradicionalmente utilizadas na resolução dos problemas de roteirização e programação de veículos, aliadas a técnica conhecida como *Cluster Search* desenvolvida por Oliveira e Lorena (2004). O *Cluster Search* é considerado uma metaheurística híbrida, pois sua estrutura é formada por uma combinação de metaheurísticas, técnicas de agrupamento e métodos de busca local. O objetivo principal do *Cluster Search* é agilizar o processo de convergência para a melhor solução associado a uma redução do esforço computacional. A escolha deste método foi impulsionada pelos bons resultados na resolução de problemas correlatos ao *DARP*, ou seja, pertencentes à classe *NP-Difícil*.

Ao optar pela implementação do *Cluster Search* (CS) observou-se que pelas características do mesmo, seria necessário a escolha de outros métodos para o emprego adequado na resolução do *DARP*. A partir de então, optou-se pela utilização da metaheurística *Simulated Annealing* (SA) como gerador de soluções para o processo de agrupamento. Com a combinação dos métodos CS-SA foi possível obter resultados significativos na resolução das instâncias apresentadas por Cordeau e Laporte (2003b).

Ao comparar as soluções obtidas com os melhores resultados disponíveis na literatura, percebe-se que os métodos CS-SA utilizados neste trabalho apresentaram resultados bastante promissores, principalmente, quando o objetivo principal é minimização dos requisitos “não essenciais”, ou seja, a redução das “inconveniências” dos clientes. Como forma de avaliar a qualidade das soluções obtidas, foi feita uma comparação com três outras abordagens encontradas na literatura, a Busca Tabu de Cordeau e Laporte, (2003b); o Algoritmo Genético de Jorgensen *et al.* (2007) e o *Simulated Annealing* de Mauri e Lorena (2009). Observou-se nessa avaliação, que em praticamente todos os casos, as soluções obtidas pelos métodos CS-SA foram melhores.

Analisou-se ainda, aspectos relacionados à eficiência e a eficácia dos métodos CS-SA para resolver o *DARP*. Em relação à eficácia foram encontrados bons resultados, quando comparados com métodos tradicionais encontrados na literatura. Já em relação à eficiência, os resultados obtidos mostram que o CS-SA foi capaz de gerar soluções de boa qualidade para todas as instâncias, em tempos computacionais expressivamente baixos.

Observando os resultados apresentados na Seção 4.1, é possível afirmar que, quando o objetivo principal do modelo estiver relacionado com minimização dos requisitos “não

essenciais” do problema, ou seja, maximização da qualidade dos serviços prestados, o *CS-SA* é uma abordagem interessante, visto que as soluções obtidas demonstram reduções significativas nos aspectos relacionados com tempo médio de viagem e o tempo médio de espera dos clientes. Vale ressaltar ainda que, no caso do *DARP*, a qualidade de serviço é uma característica de fundamental importância.

Os resultados permitem afirmar que a aplicação de métodos heurísticos para resolver o *DARP* possibilita encontrar soluções boas e até mesmo ótimas, a um custo computacional (tempo de processamento) cada vez menor.

Pode-se dizer também, que tanto o modelo matemático proposto por Mauri e Lorena (2009) quanto os métodos *CS-SA* implementados nesse trabalho, são perfeitamente aplicáveis a casos práticos, pois, as instâncias utilizadas representam casos muito próximos dos problemas encontrados no mundo real.

Com o intuito de melhorar ainda mais o desempenho do *CS-SA*, várias estratégias podem ser adotadas, como por exemplo:

- Verificar novos valores para os parâmetros (calibração);
- Implementar novas formas para gerar as soluções “centro” dos *clusters* iniciais, pois estas foram geradas de forma completamente aleatória, ou seja, geralmente soluções muito ruins;
- Implementar novas formas de atualização dos centros dos *clusters*;
- Testar outros algoritmos de busca local; e,
- Verificar outras formas de perturbação das soluções.

Outra consideração importante pode ser a utilização do *CS-SA* para resolver problemas que considerem aspectos reais da rede viária como distância, velocidade, fluxos, congestionamentos, etc. Como este tipo de informação não está disponível nas instâncias utilizadas, não foi possível efetuar tais análises. Todavia, os resultados apresentados pelos métodos utilizados nesse trabalho são animadores e impulsionam novas pesquisas que considerem esses aspectos.

Finalmente, a avaliação dos resultados obtidos, possibilita afirmar que o método *Cluster Search* aliado ao *Simulated Annealing* pode ser uma excelente alternativa na resolução de problemas considerados de alta complexidade, principalmente, quando o objetivo é a redução dos tempos computacionais.

Referências Bibliográficas

- Attanasio, A.; Cordeau, J.-F.; Ghiani, G.; Laporte, G.: *Parallel Tabu search heuristics for the dynamic multi-vehicle dial-a-ride problem*. *Parallel Computing* 30, pp. 377-387, (2004) .
- Baugh, J. W., Kakivaya, D. K. R. e Stone, J. R.: *Intractability of the dial-a-ride problem and a multiobjective solution using simulated annealing*. *Engineering Optimization*, v. 30, n. 2, p. 91-123, 1998.
- Bergvinsdottir, K. B. *The genetic algorithm for solving the dial-a-ride problem*. (IMM 2004-37). Dissertação (Master of Science in Engineering) - Technical University of Denmark (DTU), Lyngby, 2004.
- Bodin, L. D.; Golden, B. L.; Assad, A.; Ball, M.: *Routing and Scheduling of Vehicles and Crews*. *The state of the Art. Comp. Opns. Res.* 10, 69-211, 1983.
- Brejon, S. R. C. e Belfiore, P.: *A importância do enfoque sistêmico para problemas de roteirização de veículos*. *Pesquisa & Desenvolvimento em Engenharia de Produção*, v. 5, p. 4-88, 2006. Disponível em http://www.revista-ped.unifei.edu.br/documentos/Edicao_05/n5_art05.pdf . Acesso em 13/11/2008.
- Chaves, A. A.; Miralles, C. e Lorena, L. A. N.: *Clustering Search Approach for the Assembly Line Worker Assignment and Balancing Problem*. *International Conference on Computers and Industrial Engineering ICCIE 2007, Alexandria, Egito - October 20 - 23, 2007a*.
- Chaves A. A.; Correa F. A.; Lorena, L.A.N. *Clustering Search Heuristic for the Capacitated pmedian Problem*. *Springer Advances in Software Computing Series*, 44: 136 – 143, 2007b.
- Chaves, A. A. e Lorena, L. A. N.: *Hybrid Metaheuristic for the Prize Collecting Traveling Salesman Problem*. *Springer Lecture Notes in Computer Science*, v. 4972, p. 123-134, 2008. Disponível em: <http://dx.doi.org/10.1007/978-3-540-78604-7/> acessado em 04/12/2008.

- Chaves, A. A.. *Metaheurísticas híbridas com busca por agrupamentos para problemas de otimização combinatória*. Tese (Doutorado em Computação Aplicada) - Instituto Nacional de Pesquisas Espaciais (INPE), São José dos Campos, 2009.
- Clarke, G. & Wright, J., *Scheduling of vehicles from a central depot to a number of delivery points*, Operations Research, 12, 568-581, 1964.
- Cordeau, J. F. e Laporte, G. *The Dial-a-Ride Problem (DARP): Variants, modeling issues and algorithms*. 4OR - Quarterly Journal of the Belgian, French and Italian Operations Research Societies, 1: 89-101, (2003a).
- Cordeau, J. F. e Laporte, G.. *A tabu search heuristic for the static multi-vehicle dial-a-ride problem*. Transportation Research Part B: Methodological, v. 37, n. 6, p. 579-594, 2003b.
- Cordeau, J. F.. *A branch-and-cut algorithm for the dial-a-ride problem*. Operations Research, v. 54, n. 3, p. 573-586, 2006.
- Cordeau, J. F. e Laporte, G. *The dial-a-ride problem: models and algorithms*. Annals of Operations Research, v. 153, n. 1, 29-46, 2007.
- Cunha, C. B. (2000). *Aspectos práticos da aplicação de modelos de roteirização de veículos a problemas reais*. Revista Transportes da ANPET – Associação Nacional de Pesquisa e Ensino em Transportes, v.8, n.2, p.51-74, novembro.
- Desrosiers J.; Dumas Y.; Soumis F.: *A dynamic programming solution of the large-scale single vehicle dial-a-ride problem with time windows*. American Journal of Mathematical and Management Sciences, 6: 301–25, 1986.
- Diana, M. e Dessouky, M. M. *A new regret insertion heuristic for solving large-scale dial-a-ride problems with time windows*, Transportation Research B, 38B(6), 539-557, 2004.
- Dumas, Y.; Desrosiers, J.; Soumis, F.: *The pickup and delivery problem with time windows*. European Journal of Operational Research, v. 54, n. 1, p. 7-22, 1991.

- Faria, M. V. S.. *Avaliação do Roteamento em Tempo Real em um Sistema de Atendimento a Clientes de Telefonia Fixa*. In: I Rio de Transportes, 2003, Rio de Janeiro. I Rio de Transportes. Rio de Janeiro : COPPE/UFRJ, 2003.
- Hamming, R. W.. *Error detecting and error correcting codes*. Bell System Technical Journal, v. 26, n. 2, p. 147-160, 1950.
- Jaw, J.J.; Odoni, A. R.; Psaraftis, H.N. e Wilson, N. H. M.: *A heuristic algorithm for the multi-vehicle advance request dial-a-ride problem with time windows*. Transportation Research B, v.20, n.3, p.243-257, 1986.
- Jih, W. R.; Kao, C. Y.; Hsu, F. Y. J.: *Using family competition genetic algorithm in pickup and delivery problem with time window constraints*. In: IEEE INTERNATIONAL SYMPOSIUM ON INTELLIGENT CONTROL, Vancouver. Proceedings... Vancouver: IEEE, 2002. p. 496-501, 2002.
- Jorgensen, R. M.; Larsen, J.; Bergvinsdottir, K. B.: *Solving the dial-a-ride problem using genetic algorithms*. Journal of the Operational Research Society, v. 58, n. 10, p. 1321-1331, 2007.
- Kirkpatrick, S.; Gellat, D. C.; Vecchi, M. P.: *Optimization by Simulated Annealing*. Science, v. 220, n. 4598, p. 671-680, 1983.
- Mauri, G. R.. *Novas abordagens para representação e obtenção de limitantes e soluções para alguns problemas de otimização combinatória*. Tese (Doutorado em Computação Aplicada) - Instituto Nacional de Pesquisas Espaciais (INPE), São José dos Campos, 2008.
- Mauri, G.R. e Lorena, L.A.N. *Uma nova abordagem para o problema dial-a-ride*. Produção, v. 19, n. 1, p. 41-54, 2009.
- Madsen, O. B. G.; Ravn, H. F.; Rygaard, J. M.: *A heuristic for a dial-a-ride problem with time windows, multiple capacities, and multiple objectives*. Annals of Operations Research, v.60, p.193-208, 1995.

- Melachrinoudis, E.; Ilhan, A. B.; Min, H.: *A dial-a-ride problem for client transportation in a health-care organization*. *Computers & Operations Research*, 34, 742–759, 2007.
- Metropolis, N.; A. Rosenbluth, M.; Rosenbluth, A. Teller; Teller E.: *Equation of State Calculations by Fast Computing Machines*. *J. Chem. Phys.*, 21, 6, 1087-1092, 1953.
- Novaes, A. G. *Sistemas logísticos: Transporte, armazenagem e distribuição física de produtos*. São Paulo: Edgar Blücher. 372p, 1989.
- OLIVEIRA, A. C. M. *Algoritmos evolutivos híbridos com detecção de regiões promissoras em espaços de busca contínuo e discreto*. Tese (Doutorado em Computação Aplicada) - Instituto Nacional de Pesquisas Espaciais (INPE), São José dos Campos, 2004.
- Oliveira, A. C. M. e Lorena, L. A. N.: *Detecting promising areas by evolutionary clustering search*. *Advances in Artificial Intelligence*. Springer Lecture Notes in Artificial Intelligence Series, p. 385–394, 2004.
- Ronen, D.. *Perspectives on practical aspects of truck routing and scheduling*. *European Journal of Operational Research*, v.35, p.137-145, 1988.
- Solomon, M.M. *Algorithms for the vehicle routing and scheduling problems with time windows constraints*. *Operations Research*, 35, 254-265, 1987.
- Toth P. e Vigo D.. *Fast local search algorithms for the handicapped persons transportation problem*, in: I.H. Osman, J.P. Kelly (Eds.), *Meta-Heuristics: Theory and Applications*, Kluwer, Boston, pp. 677–690, 1996.
- Toth, P. e D. Vigo, D.. *Heuristic algorithms for the handicapped persons transportation problem*. *Transportation Science*, v.31, n.1, p.60-71, 1997.
- Znamensky, A. e Cunha, C. B. *Um modelo para o problema de roteirização e programação do transporte de deficientes*. In: CONGRESSO DE PESQUISA E ENSINO EM TRANSPORTES, 13., 1999, São Carlos. Anais... São Carlos: ANPET, p. 59-62, 1999.