

**UNIVERSIDADE DO VALE DO RIO DOS SINOS**  
**PROGRAMA INTERDISCIPLINAR DE PÓS-GRADUAÇÃO**  
**EM COMPUTAÇÃO APLICADA-PIPCA**

ETIENE POZZOBOM LAZZERIS SIMAS

**Utilizando a Busca Tabu na Resolução do**  
**Problema de Roteamento de Veículos**

São Leopoldo

2007

**UNIVERSIDADE DO VALE DO RIO DOS SINOS**  
**PROGRAMA INTERDISCIPLINAR DE PÓS-GRADUAÇÃO**  
**EM COMPUTAÇÃO APLICADA-PIPCA**

ETIENE POZZOBOM LAZZERIS SIMAS

**Utilizando a Busca Tabu na Resolução do**  
**Problema de Roteamento de Veículos**

Dissertação apresentada à Universidade do Vale do Rio dos Sinos (UNISINOS) como requisito parcial para obtenção do grau de Mestre em Computação Aplicada

Orientador: Prof. Dr. Arthur Tórgo Gómez

São Leopoldo

2007

ETIENE POZZOBOM LAZZERIS SIMAS

Utilizando a Busca Tabu na Resolução do Problema de Roteamento de Veículos

Dissertação apresentada à Universidade do Vale do Rio dos Sinos como requisito parcial para  
obtenção do título de mestre em Computação Aplicada

Aprovado em \_\_\_\_ de \_\_\_\_\_ de 2007

BANCA EXAMINADORA

---

Prof. Dr. Arthur Tórgo Gómez

---

Prof. Dr. Luis Antônio Nogueira Lorena

---

Prof. Dr. José Vicente Canto dos Santos

## **AGRADECIMENTOS**

Ao meu orientador Prof. Dr.Arthur Tórgo Gómez pelo incentivo na carreira acadêmica e apoio nesta caminhada.

À CAPES/PROSUP pela bolsa de mestrado.

Ao meu marido pela compreensão, paciência e companheirismo.

Aos meus pais pelo estímulo, confiança e torcida.

Aos colegas do curso pela descontração e amizade.

## **RESUMO**

Este trabalho aborda o Problema de Roteamento de Veículos em que apenas a restrição da capacidade do veículo é considerada. O objetivo deste problema é atender a uma rede de clientes, através da criação de rotas de custo mínimo que respeitem a capacidade do veículo. Para gerar soluções para este problema uma aplicação será desenvolvida utilizando a metaheurística Busca Tabu. A Busca Tabu permite que estratégias de intensificação e diversificação sejam utilizadas para guiar a procura de boas soluções. Para esta aplicação dois tipos de movimentos geradores de vizinhança serão utilizados: a realocação de vértices e a troca de vértices entre rotas. Os resultados obtidos nesta aplicação serão comparados com resultados de heurísticas clássicas e com resultados publicados por outros autores que utilizam a Busca Tabu na abordagem do Problema de Roteamento de Veículos.

## **ABSTRACT**

This work presents the Vehicle Routing Problem where only the vehicle capacity constraint is considered. The aim of this problem is to service a set of clients through the generation of least cost routes which respect the vehicle capacity constraint. To create the solutions of this problem an application will be developed using the Tabu Search meta-heuristics. The Tabu Search allows that intensification and diversification strategies guide the search of good solutions. In this application two types of movements to generate the neighborhood were defined: the vertex relocation and the vertex exchange between routes. The results obtained in this application will be compared with the results obtained by classical heuristics and by results obtained from other papers that use Tabu Search to approach the Vehicle Routing Problem.

## LISTA DE FIGURAS

FIGURA 1 - Representação de uma entrada e saída para o PCV. Fonte: Skiena (1997)..	32
FIGURA 2 - Representação do $m$ -PCV. Fonte: Bodin, Golden e Assad (1983).....	35
FIGURA 3 - Partição em subconjuntos. _____ $k$ -degree center tree( $k=5$ ),----- $y$ arestas adjacentes ao deposito ( $y=3$ ), ..... $M-y = 1$ aresta não adjacentes ao depósito. Fonte: Laporte (1992b) .....	45
FIGURA 4 - Representação da economia atingida com a utilização de um mesmo veículo para atender os vértices $i$ e $j$ . Fonte: Goldberg e Luna (1998).....	51
FIGURA 5 - Ilustração dos <i>clusters</i> gerados pelo método das varreduras. Fonte: Bodin, Golden e Assad (1983). .....	53
FIGURA 6- Ilustração do método de otimização dentro da rota. Fonte: Thangiah e Petrovic (1997). .....	54
FIGURA 7 - Ilustração do método de otimização entre rotas. Fonte: Thangiah e Petrovic (1997).....	55
FIGURA 8 - Exemplo do movimento <i>String Cross</i> . Fonte: Breedam (2001).....	55
FIGURA 9 - Exemplo do movimento <i>String Exchange</i> . Fonte: Breedam (2001).....	55
FIGURA 10- Exemplo do movimento <i>String Relocation</i> . Fonte: Breedam (2001).....	56
FIGURA 11 - Processo de busca continuado além dos ótimos locais. Fonte: Osman e Laporte (1996). .....	59
FIGURA 12 - Componentes da estratégia de Busca. Fonte: Viana (1998).....	60

FIGURA 14 – Ilustração de uma possível solução do PRV: um conjunto de rotas que atende todos os clientes. ....	68
FIGURA 15 - Arquitetura do Modelo. ....	70
FIGURA 16 – Matriz de Distância. ....	72
FIGURA 17a – Rede com a Distância entre o depósito e todos os clientes. ....	73
FIGURA 17b –Início da formação de uma rota a partir da seleção do cliente mais próximo ao depósito. ....	74
FIGURA 17c – Adição de um segundo cliente à rota. ....	74
FIGURA 17d – Adição de um terceiro cliente à rota. ....	75
FIGURA 17e – Retorno ao depósito do veículo que atende a rota e início da formação de uma nova rota. ....	75
FIGURA 18 - Estrutura geral do algoritmo da Pesquisa Tabu implementado no modelo proposto neste trabalho. ....	77
FIGURA 19 – Detalhamento do algoritmo da Pesquisa Tabu com a expansão de V1. ....	78
FIGURA 20a – Dinâmica de V1: Seleção do vértice A na rota origem R1 e do primeiro vértice a ser trocado na rota destino R4. ....	80
FIGURA 20b - Dinâmica de V1: Troca entre os vértice A e B. ....	81
FIGURA 20c - Dinâmica de V1: Seleção do vértice A na rota origem R1 e do segundo vértice a ser trocado na rota destino R4. ....	81
FIGURA 20d - Dinâmica de V1: Troca entre os vértice A e C. ....	82



FIGURA 21a – Dinâmica de V2: seleção da rota origem R1 e do vértice A. ....	83
FIGURA 21b - Dinâmica de V1: seleção de uma rota destino, R4, e inserção de A na primeira posição possível. ....	83
FIGURA 21c - Dinâmica de V2: inserção de A na segunda posição possível da rota R4. ....	84
FIGURA 21d - Dinâmica de V2: inserção de A na terceira posição possível da rota R4. ....	84
FIGURA 24 - Estrutura de armazenamento dos resultados dos experimentos. ....	90
FIGURA 22a - Variação do valor do Nbmax para cada tamanho da Lista Tabu. ....	91
FIGURA 22b - Possíveis melhores resultados analisados sob perspectiva 1. ....	92
FIGURA 23a - Variação do valor do tamanho da Lista Tabu para o Nbmax. ....	93
FIGURA 23b - Possíveis melhores resultados analisados sob perspectiva 2. ....	94
FIGURA 24 - Variação do Custo para o problema 1 usando V1 sem intensificação. ....	96
FIGURA 25 - Variação do Custo para o problema 1 usando V1 com intensificação. ....	96
FIGURA 26 - Variação do Custo para o problema 2 usando V1 sem intensificação. ....	97
FIGURA 27 - Variação do Custo para o problema 2 usando V1 com intensificação. ....	98
FIGURA 28 - Variação do Custo para o problema 3 usando V2 com intensificação. ....	98
FIGURA 29 - Variação do Custo para o problema 3 usando V2 com intensificação. ....	99
FIGURA 30 - Variação do Custo para o problema 4 usando V1 sem intensificação. ....	100
FIGURA 31 - Variação do Custo para o problema 4 usando V2 com intensificação. ....	100

FIGURA 32 - Variação do Custo para o problema 5 usando V1,V2 sem intensificação.	101
FIGURA 33 - Variação do Custo para o problema 5 usando V1,V2 com intensificação.....	102
FIGURA 34 - Processo de busca usando V1 e Lista Tabu = 200 para o problema 1.....	109
FIGURA 35 - Processo de busca usando V2 e Lista Tabu = 10 para o problema 1.....	110
FIGURA 36 - Processo de busca usando V1 e Lista Tabu = 50 para o problema 2.....	110
FIGURA 37 - Processo de busca usando V1,V2 e Lista Tabu = 100 para o problema 2.	111
FIGURA 38 - Processo de busca usando V1 e Lista Tabu = 200 para o problema 3.....	112
FIGURA 39 - Processo de busca usando V1 e Lista Tabu = 200 para o problema 3.....	112
FIGURA 40 - Processo de busca usando V2 e Lista Tabu = 200 para o problema 4.....	113
FIGURA 41 - Processo de busca usando V1,V2 e Lista Tabu = 200 para o problema 4.	114
FIGURA 42 - Processo de busca usando V1 e Lista Tabu = 50 para o problema 5.....	114
FIGURA 43 - Processo de busca usando V2 e Lista Tabu = 200 para o problema 5.....	115
FIGURA 44 - Quantidade de resultados gerados com e sem intensificação para o problema 1. ....	116
FIGURA 45 - Quantidade de resultados gerados com e sem intensificação para o problema 2. ....	116
FIGURA 46 - Quantidade de resultados gerados com e sem intensificação para o problema 3. ....	117

FIGURA 47 - Quantidade de resultados gerados com e sem intensificação para o problema 4. ....	117
FIGURA 48 - Quantidade de resultados gerados com e sem intensificação para o problema 5. ....	118
FIGURA 49 - Melhoria gerada em função do uso da estratégia de intensificação para o problema 1. ....	120
FIGURA 50 - Melhoria gerada em função do uso da estratégia de intensificação para o problema 2. ....	121
FIGURA 51 - Melhoria gerada em função do uso da estratégia de intensificação para o problema 3. ....	121
FIGURA 52 - Melhoria gerada em função do uso da estratégia de intensificação para o problema 4. ....	122
FIGURA 53 - Melhoria gerada em função do uso da estratégia de intensificação para o problema 5. ....	123
FIGURA 54 – Melhores resultados obtidos para o problema 1. ....	130
FIGURA 55– Melhores resultados obtidos para o problema 2. ....	130
FIGURA 56 – Melhores resultados obtidos para o problema 2. ....	131
FIGURA 57 – Melhores resultados obtidos para o problema 4. ....	131
FIGURA 58 – Melhores resultados obtidos para o problema 5. ....	132
FIGURA 59 – Comparação dos resultados para o problema 1. ....	137

FIGURA 60 – Comparação dos resultados para o problema 2. ....	137
FIGURA 61 – Comparação dos resultados para o problema 3. ....	138
FIGURA 62 – Comparação dos resultados para o problema 4. ....	138
FIGURA 63 – Comparação dos resultados para o problema 5. ....	139

## LISTA DE TABELAS

TABELA 1 - Relação das características dos problemas de roteamento e escalonamento. Fonte: Bodin, Golden e Assad (1983). .....	26
TABELA 2 - Relação dos valores para o Tamanho da Lista Tabu utilizados.....	88
TABELA 3- Relação dos valores para o <i>Nbmax</i> utilizado. ....	89
TABELA 4 – Combinação de movimentos utilizados. ....	89
TABELA 5 – Resultados das Médias Superiores e Inferiores. ....	102
TABELA 6 - Quantidade de melhores resultados encontrados para variação proposta do <i>Nbmax</i> aplicado ao problema 1. ....	105
TABELA 7 - Quantidade de melhores resultados encontrados para variação proposta do <i>Nbmax</i> aplicado ao problema 2. ....	105
TABELA 8 - Quantidade de melhores resultados encontrados para variação proposta do <i>Nbmax</i> aplicado ao problema 3. ....	105
TABELA 9 - Quantidade de melhores resultados encontrados para variação proposta do <i>Nbmax</i> aplicado ao problema 4. ....	106
TABELA 10 - Quantidade de melhores resultados encontrados para variação proposta do <i>Nbmax</i> aplicado ao problema 5. ....	106
TABELA 11 - Quantidade de melhores resultados para tamanho de lista menor e maior que 75 para problema 1.....	107

TABELA 12 - Quantidade de melhores resultados para tamanho de lista menor e maior que 75 para problema 2.....	107
TABELA 13 - Quantidade de melhores resultados para tamanho de lista menor e maior que 75 para problema 3.....	107
TABELA 14 - Quantidade de melhores resultados para tamanho de lista menor e maior que 75 para problema 4.....	108
TABELA 15 - Quantidade de melhores resultados para tamanho de lista menor e maior que 75 para problema 5.....	108
TABELA 16 - Quantidade de melhores resultados encontrados para o problema 1.....	119
TABELA 17 - Quantidade de melhores resultados encontrados para o problema 2.....	119
TABELA 18 - Quantidade de melhores resultados encontrados para o problema 3.....	119
TABELA 19 - Quantidade de melhores resultados encontrados para o problema 4.....	119
TABELA 20- Quantidade de melhores resultados encontrados para o problema 5.....	119
TABELA 21- Tamanho Médio e Desvio Padrão dos tamanhos de vizinhanças.....	123
TABELA 22 - Melhores resultados para o problema 1.....	127
TABELA 23 - Melhores resultados para o problema 2.....	127
TABELA 24 - Melhores resultados para o problema 3.....	128
TABELA 25 - Melhores resultados para o problema 4.....	128
TABELA 26 - Melhores resultados para o problema 5.....	129

TABELA 27 – Comparação dos resultados para os problemas 1,2 e 3 com trabalhos de Busca Tabu. ....	134
TABELA 28 – Comparação dos resultados para os problemas 4 e 5 com trabalhos de Busca Tabu. ....	135
TABELA 29 – Comparação dos resultados para os problemas 1,2 e 3 com heurísticas clássicas. ....	136
TABELA 30 – Comparação dos resultados para os problemas 4 e 5 com heurísticas clássicas. ....	136

## LISTA DE SÍMBOLOS

S – Subconjunto de vértices

B – Subconjunto de vértices utilizados para garantir eliminação de subtour

V – Conjunto de vértices

$\Phi$  - Circuito hamiltoniano

$N_v$  – número de vizinhos gerados pela Busca Tabu

$S_j^*$  - melhor vizinho da iteração j

$N(s)$  – vizinhança de uma solução tabu

Nbmax – número máximo de iterações executadas pela Busca Tabu



## **LISTA DE SIGLAS**

PRV – Problema de Roteamento de Veículos

PRVC – Problema de Roteamento de Veículos Capacitado

PCV – Problema do Caixeiro Viajante

m-PCV – Problema do m-Caixeiro Viajante

k-DCT – k-degree center tree

## SUMÁRIO

<b>1</b>	<b><u>INTRODUÇÃO</u></b>	<b>20</b>
<b>2</b>	<b><u>O PROBLEMA DE ROTEAMENTO DE VEÍCULOS</u></b>	<b>23</b>
2.1	CONTEXTUALIZAÇÃO DO PROBLEMA	23
<b>3</b>	<b><u>DEFINIÇÃO CLÁSSICA E PROBLEMAS RELACIONADOS</u></b>	<b>29</b>
3.1	DEFINIÇÃO CLÁSSICA DO PROBLEMA	29
3.2	O PRV RELACIONADO A OUTROS PROBLEMAS	31
3.2.1	PROBLEMA DO CAIXEIRO VIAJANTE (PCV)	31
3.2.2	PROBLEMA DO M-CAIXEIRO VIAJANTE (M-PCV)	35
<b>4</b>	<b><u>REVISÃO DO ESTADO DA ARTE</u></b>	<b>40</b>
4.1	ESTADO DA ARTE SOBRE OS MÉTODOS PARA RESOLUÇÃO DO PRV	40
4.1.1	MÉTODOS EXATOS PARA RESOLUÇÃO DO PRV	42
4.1.2	HEURÍSTICAS CLÁSSICAS PARA A RESOLUÇÃO DO PRV	50
4.1.3	METAHEURÍSTICAS	57
<b>5</b>	<b><u>MODELO PROPOSTO</u></b>	<b>67</b>
5.1	DEFINIÇÃO DO MODELO	67
5.2	FORMULAÇÃO MATEMÁTICA	68
5.3	ARQUITETURA DO MODELO	70
5.3.1	MÓDULO DE GERAÇÃO DA REDE	71
5.3.2	MÓDULO DA GERAÇÃO DA SOLUÇÃO INICIAL	72
5.3.3	MÓDULO DA BUSCA TABU	76

5.3.4	MÓDULO DA VALIDAÇÃO .....	86
<b>6</b>	<b><u>EXPERIMENTOS E VALIDAÇÃO DO MODELO.....</u></b>	<b>87</b>
<b>6.1</b>	<b>PLANEJAMENTO DOS EXPERIMENTOS.....</b>	<b>87</b>
<b>6.2</b>	<b>RESULTADOS OBTIDOS.....</b>	<b>94</b>
6.2.1	ANÁLISE SOB PERSPECTIVA 1 – VARIAÇÃO DO NBMAX PARA CADA TAMANHO DA LISTA TABU...95	
6.2.2	ANÁLISE SOB À PERSPECTIVA 2 - VARIAÇÃO DO TAMANHO DA LISTA TABU PARA NBMAX .....106	
6.2.3	ANÁLISE COMPARATIVA DO PROCESSO DE BUSCA COM USO DA ESTRATÉGIA DE INTENSIFICAÇÃO E SEM O USO DA ESTRATÉGIA DE INTENSIFICAÇÃO .....108	
<b>6.3</b>	<b>VALIDAÇÃO.....</b>	<b>132</b>
<b>7</b>	<b><u>CONSIDERAÇÕES FINAIS .....</u></b>	<b>140</b>
<b>8</b>	<b><u>REFERÊNCIAS BIBLIOGRÁFICAS.....</u></b>	<b>144</b>

# 1 INTRODUÇÃO

Nos dias de hoje, a comunidade científica vem fazendo um grande esforço no sentido de trabalhar na busca de algoritmos eficientes que possam ser utilizados na resolução de problemas de Otimização Combinatória. Na área da Otimização Combinatória existem diversos problemas de aplicação prática que necessitam de algoritmos eficientes para que possam ser tratados de maneira aceitável computacionalmente. Alguns problemas clássicos como o Problema do Caixeiro Viajante, o Problema da Mochila e o Problema da Coloração de Grafos vêm sendo utilizados como fonte de pesquisa entre os estudiosos da área. Os problemas de Otimização Combinatória geralmente são simples de serem enunciados, entretanto de difícil resolução (Osman e Laporte, 1996), pois envolvem uma grande quantidade de possíveis soluções a serem avaliadas. Para problemas deste tipo, muitas vezes, métodos exatos se tornam ineficientes, pois exigem um tempo computacional muito grande. Neste caso, as heurísticas são os métodos mais apropriados para serem usados. Embora não seja garantido que as heurísticas encontrem o resultado “ótimo”, uma vez que elas são construídas de maneira a não percorrer todo o espaço de busca, diminuindo assim o tempo necessário para encontrar uma solução, muitas vezes é preferível utilizar as soluções “boas” proporcionadas pelas heurísticas do que aceitar o tempo excessivo que os métodos exatos exigem na busca do resultado “ótimo”.

Na área da logística e gerenciamento da distribuição, freqüentemente aparece um tipo de situação que pode ser tratada como sendo um problema de Otimização Combinatória. É comum o fato de depósitos terem que suprir seus clientes ou pontos de demanda, através do transporte de insumos ou serviços. Nessas situações, as empresas precisam criar um modelo de roteamento, ou seja, um roteiro que distribua e ordene os clientes a serem atendidos. Como característica desse modelo, deseja-se que o atendimento seja feito da maneira mais econômica possível, considerando a distância, tempo e demais custos envolvidos. Esse tipo de situação, exposta acima, caracteriza o Problema de Roteamento de Veículos que será abordado neste trabalho. Este problema, que pode ser considerado uma extensão do Problema do Caixeiro Viajante (Toth e

Vigo, 2002; Golden, Magnanti e Nguyen, 1977), geralmente aparece em situações em que existem clientes precisando ser atendidos por veículos localizados em depósitos. Associados a este atendimento podem existir diversas restrições (capacidade do veículo, tempo de duração da rota, etc) e diferentes objetivos como minimizar o custo deste atendimento, a distância percorrida ou a quantidade de veículos necessários. Devido à dificuldade intrínseca deste problema, que é classificado como *NP-Hard* (Lenstra e Rinnooy Kan, 1981), a grande maioria dos métodos propostos para resolvê-los são baseados em heurísticas. A partir de 1990 os métodos conhecidos como metaheurísticas vêm ganhando destaque no meio científico em função dos resultados significativos que a eles são atribuídos. Métodos como Busca Tabu e *Simulated Annealing* vêm sendo amplamente utilizados na resolução deste problema.

O objetivo deste trabalho é gerar um modelo para abordar o problema de Roteamento de Veículos, utilizando a Busca Tabu, de modo a propor estratégias de busca no espaço aplicáveis ao problema. Pode-se dizer que, uma das principais contribuições deste trabalho se refere à análise da utilização de mecanismos flexíveis de geração de vizinhança que podem ser combinados, com o intuito de aumentar a diversidade de soluções a serem geradas. Espera-se comprovar que uma maior diversidade de soluções geradas contribua na obtenção de resultados mais satisfatórios para o problema. Uma revisão do estado da arte sobre o Problema será apresentada, além de técnicas clássicas de resolução e de trabalhos em que a Busca Tabu foi utilizada.

Este trabalho está organizado da seguinte forma: O capítulo 2 apresenta a caracterização do Problema de Roteamento de Veículos, sua contextualização, características e algumas aplicações. No capítulo 3 é apresentada uma definição clássica do problema. Também são apresentados problemas clássicos relacionados ao Problema de Roteamento de Veículos, como o Problema do Caixeiro Viajante e o problema do *m*-Caixeiro Viajante, bem como alguns métodos tradicionais de resolução dos mesmos. No capítulo 4 é feito um levantamento do estado da arte do problema, e são apresentados diversos métodos de resolução classificados como métodos exatos, heurísticas clássicas e metaheurísticas. O capítulo 5 apresenta a definição do problema

abordado neste trabalho, o modelo construído, a formulação matemática e a arquitetura da aplicação a ser construída para a resolução do modelo. O capítulo 6 apresenta o planejamento dos experimentos, os resultados obtidos e a validação utilizada no trabalho. O capítulo 7 apresenta as considerações finais e perspectivas de trabalhos futuros.

## 2 O PROBLEMA DE ROTEAMENTO DE VEÍCULOS

O objetivo deste capítulo é apresentar o Problema de Roteamento de Veículos, suas principais características e variações.

### 2.1 CONTEXTUALIZAÇÃO DO PROBLEMA

O Problema de Roteamento de Veículos (PRV) foi primeiramente introduzido por Dantzig e Ramser (1959) e tem se tornado um dos problemas mais estudados na área da Otimização Combinatória (Cordeau *et al*, 2002; Kytöjoki *et al*, 2005). Um problema de roteamento pode de maneira sucinta ser definido pela existência de um conjunto de clientes que devem ser atendidos por uma frota de veículos, respeitando-se algumas restrições definidas (Laporte, 1992a), (Xu e Kelly, 1996). Os problemas de roteamento são freqüentemente abordados no contexto da área de logística, transporte e distribuição (Ho e Haugland, 2004; Xu e Kelly, 1996). A logística pode ser, resumidamente, definida como a provisão de bens e serviços de pontos de suprimento a pontos de demanda (Bodin, Golden e Assad, 1983). Um sistema completo de logística envolve todas as atividades necessárias para administrar o fluxo de bens e serviços do ponto de onde são produzidos até o ponto aonde serão consumidos e tem como atividades primárias o transporte, a manutenção de estoques e o processamento de pedidos (Ballou, 1993). Marinakis e Migdalas (2002) dividem a cadeia de suprimento em quatro áreas que mais se destacam no que se refere ao processo de tomada de decisão: localização, produção, inventário e transporte/distribuição. E, segundo eles, as decisões de transporte podem envolver a seleção, armazenamento, roteamento e escalonamento. Os autores também afirmam que o transporte é uma área chave de decisão dentro do *mix* logístico absorvendo, na média, um alto percentual dos custos. Para Ballou (2001), estes custos variam tipicamente entre um e dois terços do total dos custos logísticos. Sendo assim, devido à representatividade do custo da atividade de transporte, muitas empresas consideram o transporte como a atividade de maior importância dentro da logística (Ballou, 2001; Barbarosoglu e Ozgur, 1999). Segundo Thangiah e Petrovic

(1997), o custo do transporte é dependente da minimização da distância total viajada pelos veículos e do número de veículos requeridos para atender as demandas. Sendo assim, a necessidade de melhorar a eficiência desta atividade é de relevante importância. A busca pela redução do custo do transporte, através da criação de um modelo de roteamento que disponibilize trajetos mais econômicos visando à minimização do tempo, distância e custos associados (Barbarosoglu e Ozgur, 1999), além da quantidade de veículos utilizados é um problema frequente de tomada de decisão na área da logística (Ballou, 2001). A importância dos problemas de roteamento é evidente devido à magnitude dos custos de distribuição associados. Nesse contexto, um pequeno percentual que consiga se reduzir do custo com transporte e distribuição representa uma significativa economia (Bodin, Golden e Assad, 1983). Uma distribuição eficiente é importante, não apenas para a sobrevivência de diversos provedores de serviços logísticos, mas também, para a competitividade econômica das regiões possibilitando redução no custo dos bens aos consumidores. Essa redução pode ser conseguida, em particular, através do uso de um modelo de rotas e escalonamento de alta qualidade para os veículos que executam essa tarefa (Kytöjoki *et al*, 2005).

O PRV é um nome dado a uma classe de problemas que envolvem a visita de clientes por veículos e no qual um custo mínimo para esse atendimento deve ser obtido (Tarantilis, Ioannou, Prastacos, 2005; Bodin, Golden e Assad, 1983). Dentro desta classe, Bodin, Golden e Assad (1983) incluem os seguintes problemas:

O *Problema do Caixeiro Viajante* que consiste em determinar o ciclo de custo mínimo que passa em todos os vértices exatamente uma vez;

O *Problema do Carteiro Chinês* que requer a determinação do ciclo de custo mínimo que passa em cada aresta uma única vez;

O *Problema do m-Caixeiro Viajante*, em que existem  $m$  veículos para os quais deve ser determinada uma seqüência de clientes a visitar, cada seqüência com no mínimo 1 cliente;



O *Problema do Roteamento com um Depósito e Múltiplos Veículos* em que existem  $m$  veículos para os quais deve ser determinada uma seqüência de clientes a visitar, entretanto obedecendo as restrições de capacidade dos veículos;

O *Problema do Roteamento com Múltiplos Depósitos e Múltiplos Veículos* cuja descrição se assemelha ao item anterior, entretanto os veículos não estão centralizados em um único depósito.

Entre esses problemas, é conhecido como a versão clássica ou versão básica desta classe (Bodin, Golden e Assad, 1983; Cordeau *et al*, 2002; Mester e Bräysy, 2005; Li, Golden, Wasil, 2005) o Problema de Roteamento com um Depósito e Múltiplos Veículos. É comum os autores (Ralph *et al*, 2001; Tarantilis, Ioannou, Prastacos, 2005; Laporte, 1992a; Thangiah a Petrovic, 1997; Toth e Vigo, 2002) abordarem este problema como sendo o Problema de Roteamento de Veículos Capacitado (PRVC), aonde a única restrição existente se refere à capacidade do veículo. Entretanto costuma-se também encontrar trabalhos na literatura com a versão clássica deste problema sendo abordada com a restrição de tempo de duração das rotas (Cordeau *et al*, 2002; Barbarasoglu e Ozgur, 1999; Mester, Bräysy, 2005), além da restrição de capacidade dos veículos.

Além das diferentes restrições que podem estar existir, este problema pode apresentar diversas características que podem ser usadas para representar variações do PRV. Bodin, Golden e Assad (Bodin, Golden e Assad, 1983) apresentam na Tabela 1 algumas características e restrições que podem estar presentes nos problemas:

**TABELA 1** - Relação das características dos problemas de roteamento e escalonamento. Fonte: Bodin, Golden e Assad (1983).

Características	Possíveis Opções
Tamanho da frota disponível	Um veículo; múltiplos veículos;
Tipo de frota disponível	Homogênea; heterogênea;
Domicílio dos Veículos	Único depósito; múltiplos depósitos;
Natureza das demandas	Estocástica; determinística;
Localização das demandas	Nos vértices; nos arcos;
Tipo de rede	Não-direcionada; direcionada; euclidiana;
Restrições quanto à capacidade dos veículos	Capacidades iguais; capacidades diferentes; sem capacidade;
Tempo Máximo de duração da rota	O mesmo tempo para todas as rotas; diferentes para rotas diferentes; não imposto;
Operação	Entrega; recolhimento; ambos;
Custos	Variáveis por rota; fixos;
Objetivos	Minimizar custo total das rotas; Minimizar a soma dos custos fixos e variáveis; minimizar número de veículos requeridos;

Tipicamente, o problema abordado é definido em um grafo não-direcionado e completo onde existe um conjunto de vértices representando os clientes e um depósito e um conjunto de arestas ligando os vértices (Barbarasoglu e Ozgur, 1999; Tarantilis, Ioannou, Prastacos, 2005; Kytöjoki *et al*, 2005; Cordeau *et al*, 2002). Para este grafo é definida uma matriz de custos associada ao conjunto de arestas que representa as distâncias ou custos entre os clientes e o depósito. A frota de veículos localizada no depósito tem capacidade homogênea e a quantidade de veículos é suficiente para atender todos os clientes. São associados pesos não-negativos a cada cliente para representar a demanda e o tempo de serviço de cada um. O Problema consiste em criar rotas para os veículos tais que (Cordeau *et al*, 2002): i) cada rota inicie e termine no depósito; ii) cada cliente seja visitado uma vez por um único veículo; iii) a capacidade do veículo

não deve ser ultrapassada na rota; iv) o tempo de duração da rota não deve ultrapassar um limite definido e v) o custo total da rota seja minimizado.

Geralmente, os custos entre cada par de clientes são iguais em ambas direções, resultando em uma matriz de custo simétrica, quando em algumas aplicações uma restrição quanto à direção entre os clientes é imposta, o problema é dito assimétrico. Também é comum assumir que os grafos são completos (ou seja, existem arestas conectando todos os pares de vértices), uma vez que isto simplifica a notação (Toth e Vigo, 2002). Em diversas situações a matriz de custos satisfaz a desigualdade triangular,  $c_{ik} + c_{kj} \geq c_{ij}$  para todo  $i, j, k \in V$ . Neste caso, não é conveniente desviar do *link* direto entre dois vértices  $i$  e  $j$ . Em algumas instâncias os vértices são associados a pontos no plano com coordenadas dadas e o custo  $c_{ij}$ , para todos os arcos  $(i,j) \in A$ , é definido como sendo a distância Euclidiana entre dois pontos correspondendo aos vértices  $i$  e  $j$ . Nesse caso a matriz é simétrica e satisfaz a desigualdade triangular (Toth e Vigo, 2002).

O PRV possui grande aplicação principalmente nas áreas de logística e distribuição. A maioria dos problemas no domínio da distribuição de bens pode ser vista como um Problema de Roteamento de Veículos (Mester e Bräysy, 2005). Abaixo uma relação de aplicações práticas do problema que são encontradas na literatura:

- Distribuição de produtos eletrônicos (Barbarasolgu e Ozgur, 1999);
- Recolhimento de lixo (Kulcar, 1996);
- Roteamento de helicópteros (Timlin e Pulleyblank, 1990);
- Entrega de correspondência bancária (Malmborg e Simons, 1989);
- Distribuição de Manufaturados (Perls e Daskin, 1985);
- Distribuição de Alimentos (Bartholdi *et al*, 1983);

-Sistemas de transportes coletivos urbanos (Ceder e Stern, 1981);

-Distribuição de Jornais (Golden e Magnanti, 1977);

-Transporte Escolar (Newton e Thomas, 1974);

Devido à dificuldade intrínseca do problema, é comum o estudo de problemas com o seu escopo bem delimitado, ou seja, com a definição das características e restrições abordadas e orientação da solução para atender somente tais características. Neste trabalho será abordado o PRVC. Neste caso assume-se que a única restrição existente se refere à capacidade do veículo. O problema apresenta uma frota de veículos homogênea de tamanho suficiente para atender todos os clientes da rede de demanda. Todos os veículos estão alocados em um único depósito e todas as rotas devem iniciar e terminar no depósito. No capítulo a seguir, a definição do problema será vista mais detalhadamente.

### 3 DEFINIÇÃO CLÁSSICA E PROBLEMAS RELACIONADOS

Este capítulo visa apresentar a definição clássica do problema, algumas formulações matemáticas, além de apresentar os conceitos e alguns métodos de resolução do problema do Caixeiro Viajante e do problema do  $m$ -Caixeiro Viajante. Estes problemas são freqüentemente utilizados como fonte de estudo para o PRV em função de suas similaridades.

#### 3.1 DEFINIÇÃO CLÁSSICA DO PROBLEMA

A versão clássica do problema freqüentemente é definida como um grafo não-direcionado  $G = (V, A)$  onde  $V$  representa o conjunto de vértices e  $A$  o conjunto de arestas. Existe uma matriz de custos  $C_{ij}$  associada a  $A$ . O vértice  $v_0$  representa o depósito e os vértices restantes representam os clientes a serem atendidos. Uma demanda  $d_i$  é associada a cada vértice em  $V \setminus \{0\}$ . Existe uma capacidade total  $Q$  para cada veículo que deve ser respeitada. Todas as rotas devem iniciar e terminar no depósito e cada cliente deve ser atendido uma única vez. O objetivo do problema é minimizar o custo total das rotas.

Em alguns trabalhos é considerada uma variável  $T$  como o tempo de duração da rota e nesse caso, além da demanda, a cada cliente é atribuído um tempo  $t_i$  de serviço. O tempo total de duração da rota não pode exceder a  $T$ . Uma formulação típica para o problema é apresentada por Barbarosoglu e Ozgur (1999):

$$\text{Minimizar } \sum_i \sum_j \sum_v c_{ij} X_{ij}^v; \quad (3.1)$$

sujeito a:

$$\sum_i \sum_v X_{ij}^v = 1, \text{ para todo } j; \quad (3.2)$$

$$\sum_j \sum_v X_{ij}^v = 1, \text{ para todo } i; \quad (3.3)$$

$$\sum_i X_{ip}^v - \sum_j X_{pj}^v = 0, \text{ para todo } p \text{ e } v; \quad (3.4)$$

$$\sum_i d_i \left( \sum_j X_{ij}^v \right) \leq Q_v, \text{ para todo } v; \quad (3.5)$$

$$\sum_i t_i^v \sum_j X_{ij}^v + \sum_j \sum_i t_{ij}^v X_{ij}^v \leq T_v, \text{ para todo } v; \quad (3.6)$$

$$\sum_{j=1}^n X_{0j}^v \leq 1, \text{ para todo } v; \quad (3.7)$$

$$\sum_{i=1}^n X_{i0}^v \leq 1, \text{ para todo } v; \quad (3.8)$$

$$X_{ij}^v \in Z, \text{ para todo } i, j \text{ e } v; \quad (3.9)$$

onde  $X_{ij}^v$  são variáveis binárias que indicam se o arco  $(v_i, v_j)$  é utilizado pelo veículo  $v$ . A função objetivo de minimização da distância/custo/tempo aparece na equação (3.1). As restrições representadas nas equações (3.2) e (3.3) juntas garantem que cada vértice é atendido por um único veículo. A equação (3.4) garante que um veículo deixa o vértice aonde foi atender a demanda tão logo alcance este vértice. A capacidade do veículo  $Q_v$  é expressa na equação (3.5) considerando que o limite máximo da duração da rota é dado pela equação (3.6). As restrições (3.7) e (3.8) expressam que a disponibilidade do veículo não pode ser excedida. A equação (3.9) representa a eliminação de *subtour* onde  $Z$  pode ser definido por:

$$Z = \left\{ X_{ij}^v \right\} : \sum_{i \in B} \sum_{j \in B} X_{ij}^v \leq |B| - 1 \quad \text{para } B \subseteq V \setminus \{0\}; |B| \geq 2.$$

Outras formulações para o problema podem ser encontradas nos trabalhos de Ralphs *et al* (2003), Toth e Vigo (2002) e Golden, Magnanti e Nguyen (1977).

### 3.2 O PRV RELACIONADO A OUTROS PROBLEMAS

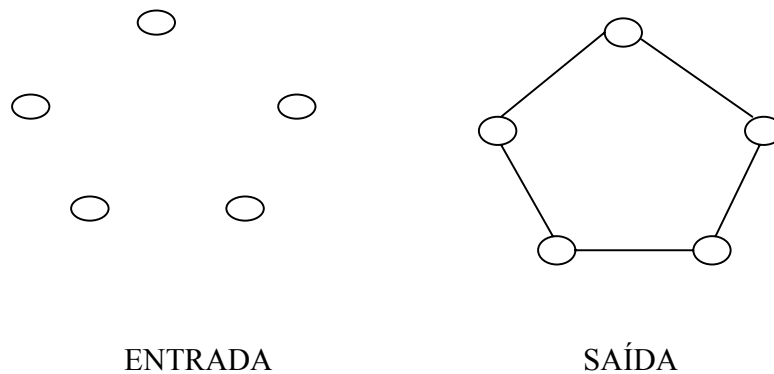
Freqüentemente o PRV é relacionado a outros problemas de Otimização Combinatória. É comum a associação com o Problema do Caixeiro Viajante (PCV) e com o Problema do  $m$ -Caixeiro Viajante ( $m$ -PCV). Esse tipo de associação entre os problemas permite que as abordagens de resolução e formulação para o PCV e para o  $m$ -PCV sejam válidas e aplicadas também na resolução do PRV. A seguir será apresentada uma revisão sobre os problemas citados, abordando sua definição, algumas formulações e métodos de resolução.

#### 3.2.1 Problema do Caixeiro Viajante (PCV)

Alguns autores (Toth e Vigo, 2002; Golden, Magnanti e Nguyen, 1977; Christofides, Mingozzi e Toth, 1981) abordam o PRV como extensão do clássico Problema do Caixeiro Viajante (PCV). Basicamente, este problema pode ser definido como sendo o problema de determinar o ciclo de custo mínimo que passa por todos os vértices uma única vez (Toth e Vigo, 2002). Este ciclo é freqüentemente denominado de *tour* na literatura. Na teoria dos grafos este problema pode ser definido sobre um grafo  $G = (V, A)$  com  $n$  vértices  $V = \{v_1, \dots, v_n\}$  e um conjunto de arestas  $A = \{(v_i, v_j) | v_i, v_j \in V, i \neq j\}$  com uma matriz de custos (ou distância) não negativa  $C = (c_{ij})$  associada a  $A$ . Um ciclo que passa em todos os vértices uma única vez é denominado ciclo Hamiltoniano. Então, o objetivo do PCV é encontrar o ciclo Hamiltoniano de menor custo (Bektas, 2006). Desta maneira, o Problema do Caixeiro Viajante pode ser interpretado como um Problema de Roteamento de Veículos com 1 depósito e 1 veículo cuja capacidade excede a demanda total (Golden, Magnanti e Nguyen, 1977). Para uma apresentação

mais detalhada sobre o PCV é sugerida a leitura de Bellmore e Nemhauser (1968), Christofides (1979) e Lawler *et al* (1985).

A figura 1 ilustra um exemplo de entrada e saída para o PCV. Como entrada é apresentado um conjunto de vértices, o objetivo do problema é unir todos os vértices formando o Ciclo Hamiltoniano. Como saída é apresentado uma *tour* ou ciclo Hamiltoniano obtido.



**FIGURA 1** - Representação de uma entrada e saída para o PCV. Fonte: Skiena (1997).

Laporte e Nobert (1980) apresentam a formulação inteira para o PCV proposta por Dantzig *et al* (1954), onde:

$N$  = número de vértices na rede;

$x_{ij} = 1$  se a aresta  $(i,j)$  está na *tour* e 0 caso contrário;

$c_{ij}$  = custo da aresta  $(i,j)$ ;



$$\text{Minimizar } \sum_{i,j \in N} c_{ij} x_{ij}; \quad (3.18)$$

sujeito a:

$$\sum_{i \in N} x_{ij} = 1, (j \in N); \quad (3.19)$$

$$\sum_{j \in N} x_{ij} = 1, (i \in N); \quad (3.20)$$

$$\sum_{i \in S, j \in S^c} x_{ij} \geq 1, (2 \leq |S| \leq n-2, S \subset N); \quad (3.21)$$

$$x_{ij} = 0 \text{ ou } 1, (i \neq j, i, j \in N); \quad (3.22)$$

onde a equação (3.18) representa a função objetivo. As equações (3.19), (3.20) e (3.22) descrevem um problema de designação e a equação (3.21) garante que o grafo é conexo.

### 3.2.1.1 Métodos para resolução do PCV

Existem diversas técnicas exatas para resolução do PCV, entre os métodos exatos, segundo Christofides et al (1979a) grande parte dos métodos são algoritmos de *branch-and-bound* com os limites inferiores sendo calculados através de relaxações do problema original. Quando existe um problema difícil a ser resolvido e que não se sabe como resolvê-lo, um dos possíveis procedimentos a ser feito é relaxar suas restrições até que se tenha um problema que se saiba como resolver eficientemente (Cook et al, 1998). Laporte (1992b) e Christofides (1979a) apresentam algumas formulações para o PCV e algoritmos de *branch-and-bound* com limites inferiores calculados a partir de relaxações baseadas em problemas de designação, árvores de expansão mínima e *2-matching*. Uma vez que existem algoritmos eficientes para resolver estes

problemas, limites inferiores para o PCV podem ser facilmente obtidos. Christofides (1979a) aborda o cálculo de um limite inferior para o PCV através da resolução da árvore de expansão mínima da seguinte maneira: Supondo um grafo não direcionado  $G$  e  $\Phi$  um circuito hamiltoniano que é a solução do PCV. Seja  $v$  um determinado vértice em  $\Phi$ . Se dois arcos incidentes em  $v$  forem removidos, o grafo restante é o caminho  $P$ , construído com os vértices de  $G - v$ . Se  $T$  for a árvore de expansão mínima destes vértices, o custo de  $T$  é um limite inferior para o custo de  $P$ . Supondo que os dois arcos que foram removidos tenham um tamanho correspondente à soma dos dois menores vértices incidentes de  $v$ , o custo dos arcos de  $T$  somado ao custo dos arcos removidos de  $v$  é um limite inferior para  $\Phi$ .

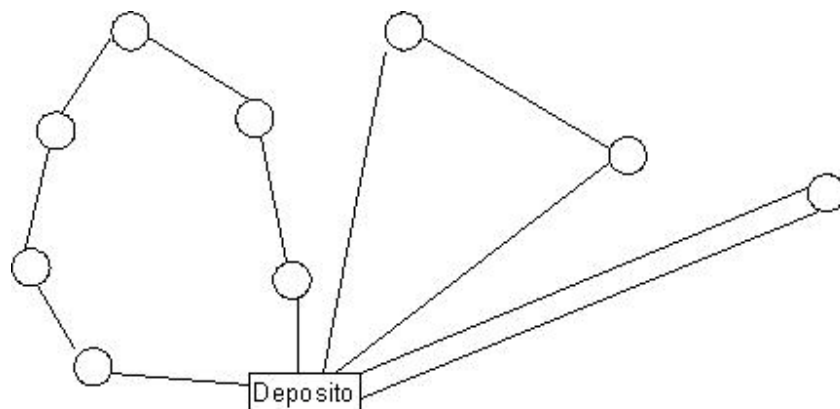
Com relação aos métodos heurísticos, é possível afirmar que se baseiam, principalmente, na construção de rotas. O Algoritmo do Vizinho mais Próximo é um método simples e rápido de implementar. A idéia é construir uma *tour* sempre adicionando o vértice mais próximo ao último que foi inserido. Esse procedimento segue continuamente até que todos os vértices tenham sido adicionados. Os Métodos de Inserção iniciam uma *tour* unindo dois vértices e então adicionando os vértices seguintes um a um, de acordo com uma política especificada. Algumas políticas frequentemente usadas que podem ser citadas são: Inserção do mais Próximo, que a cada passo escolhe um vértice cujo custo até qualquer vértice na rota seja mínimo e Inserção do mais Distante, que a cada passo insere na *tour* o vértice mais distante.

Entre os Métodos de Melhoramento de rota, segundo Cook *et al* (1998) o mais simples é o chamado 2-opt. Ele considera cada par não adjacente de arestas na *tour* por vez. Se essas arestas são excluídas, então a *tour* se divide em dois caminhos. Existe então uma única maneira de unir estes dois caminhos e formar uma nova *tour*. Se o custo da nova *tour* formada for melhor que o custo da *tour* original, a *tour* atual substitui a original e o processo se repete. Este procedimento 2-opt pode ser generalizado para um procedimento k-opt onde são considerados todos os subconjuntos de arestas da *tour* de tamanho  $k$ .

Lin e Kernighan (1973) propõem um método baseado no k-opt apresentando duas novas características: o valor de k pode variar e quando um melhoramento é encontrado não é necessário que ele seja executado imediatamente. A busca pode continuar na esperança de encontrar um melhoramento ainda maior.

### 3.2.2 Problema do m-Caixeiro Viajante (*m*-PCV)

O trabalho de Golden, Magnanti e Nguyen (1977), Bektas (2006) e Bodin *et al* (1983) associam o Problema do *m*-Caixeiro Viajante (*m*-PCV) ao PRV. Este problema é uma generalização do PCV e pode ser definido como: Dado um conjunto de vértices em que existe um conjunto de caixeiros viajantes, deixe *m* ser a quantidade de caixeiros localizados em um único vértice depósito. Os vértices restantes são clientes a serem visitados. Então, o *m*-Caixeiro Viajante consiste em encontrar *tours* para todos os *m* caixeiros, que iniciem e terminem no depósito, tal que cada cliente seja visitado apenas uma vez e o custo total seja minimizado (Bektas, 2006). Desta maneira, o *m*-PCV pode ser considerado como uma relaxação do PRV, uma vez que as restrições de capacidade dos veículos foram removidas (Bodin, Golden e Assad, 1983). A figura 2 a seguir ilustra uma possível solução para um Problema de Roteamento de Veículos.



**FIGURA 2** - Representação do *m*-PCV. Fonte: Bodin, Golden e Assad (1983).

Bektas (2006) apresenta diversas formulações de Programação Inteira para o  $m$ -PCV. A seguir, serão apresentadas duas formulações para o problema: (a) baseada em designação e (b) a formulação de Laporte e Nobert (1980). Antes de apresentar as formulações algumas definições são necessárias:

O  $m$ -PCV é definido em um grafo  $G = (V,A)$ , onde  $V$  é o conjunto de  $n$  vértices e  $A$  o conjunto de arestas. Seja  $C = (c_{ij})$  uma matriz de custo associada com  $A$ . A matriz  $C$  é dita simétrica quando  $c_{ij} = c_{ji}$ ,  $\forall (i,j) \in A$  e assimétrica caso contrário.  $X_{ij}$  é uma variável binária que assume o valor 1 se o arco  $(i,j)$  é usado na *tour* e 0, caso contrário.

(a) Formulação de programação inteira baseada em designação:

$$\text{Minimizar } \sum_{i=1}^n \sum_{j=1}^n c_{ij} x_{ij}; \quad (3.23)$$

sujeito a:

$$\sum_{j=2}^n x_{1j} = m; \quad (3.24)$$

$$\sum_{j=2}^n x_{j1} = m; \quad (3.25)$$

$$\sum_{j=2}^n x_{1j} = 1, j = 2, \dots, n; \quad (3.26)$$

$$\sum_{j=1}^n x_{1j} = 1, i = 2, \dots, n; \quad (3.27)$$

$$+ \text{restrições de eliminação de } \textit{subtour} \quad (3.28)$$

$$x_{ij} \in \{0,1\}, \forall (i,j) \in A; \quad (3.29)$$

onde as equações (3.26), (3.27) e (3.29) são as típicas restrições de designação e (3.24) e (3.25) garantem que exatamente  $m$  veículos partem e retornam ao vértice 1 (depósito). A restrição (3.28) previne *subtours* e são chamadas de Restrições de eliminação de *subtour* (SECs). Diversas SECs têm sido propostas na literatura. Bektas (2006) destaca as restrições propostas por Dantzig *et al* (1954). Estas restrições são mostradas a seguir:

$$\sum_{i \in S} \sum_{j \in S} x_{ij} \leq |S|, \quad \forall S \subseteq V \setminus \{1\}, S \neq \emptyset \quad (3.30)$$

ou

$$\sum_{i \in S} \sum_{j \in S} x_{ij} \geq 1, \quad \forall S \subseteq V \setminus \{1\}, S \neq \emptyset. \quad (3.31)$$

As restrições (3.30) e (3.31) impõem requerimentos de conectividade para a solução, prevenindo a formação de *subtours* de cardinalidade  $S$  não incluindo o depósito.

(b) Formulação de Laporte e Nobert (1980)

O trabalho de Laporte e Nobert (1980) apresenta duas formulações para o  $m$ -PCV. Ambas consideram um custo fixo  $f$  para cada caixeiro utilizado na solução. A seguir será apresentada a formulação para o caso assimétrico:

$$\text{Minimizar } \sum_{i \neq j} c_{ij} x_{ij} + fm; \quad (3.32)$$

sujeito a:

$$\sum_{j=2}^n x_{1j} = 2m; \quad (3.33)$$

$$\sum_{i \neq k} x_{ik} + \sum_{j > k} x_{kj} = 2, \quad k = 2, \dots, n; \quad (3.34)$$

$$\sum_{i \neq j, i, j \in S} x_{ij} \leq |S| - 1, 3 \leq |S| \leq n - 2, S \subseteq V \setminus \{1\}; \quad (3.35)$$

$$x_{ij} \in \{0, 1\}, 1 < i < j; \quad (3.36)$$

$$x_{1j} \in \{0, 1, 2\}, j = 2, \dots, n; \quad (3.37)$$

$$m \geq 1 \text{ e inteiro}; \quad (3.38)$$

onde a equação (3.32) representa a função objetivo que é minimizar o custo total da viagem e o número de caixeiros usados na solução. As equações (3.33) e (3.34) são as restrições de grau no vértice depósito e no restante dos vértices, respectivamente. A equação (3.35) representa as restrições de eliminação de *subtours* de Dantzig *et al* (1954). Esta formulação não é considerada uma formulação inteira puramente binária devido a variável  $x_{ij}$  que pode ser 0, 1 ou 2. A variável  $x_{ij}$  é definida apenas para  $i < j$ , uma vez que o problema é simétrico e apenas uma única variável é suficiente para representar cada aresta usada na solução.

### 3.2.2.1 Métodos para Resolução do m-PCV

Segundo Bektas (2006) as abordagens para solução deste problema são divididas em 3 grupos: métodos exatos, transformação para o PCV e heurísticas. Com relação aos métodos exatos, a primeira abordagem para resolver o *m*-PCV, sem a transformação no PCV aparece no trabalho de Laporte e Nobert (1980), que propõe dois algoritmos para resolver o problema: *straight algorithm* e *reverse algorithm*, ambos baseados na relaxação de algumas restrições do problema. O *straight algorithm* resolve o problema, relaxando as restrições de SEC e verificando se alguma das SEC é violada, no passo seguinte, uma solução inteira é obtida. Se alguma SEC for violada, a restrição é introduzida pra remover a *subtour*. No *reverse algorithm* as SEC são também relaxadas, entretanto a verificação é executada antes de se gerar uma solução inteira.

Métodos de *branch-and-bound* também foram propostos como solução para o  $m$ -PCV nos trabalhos de Ali e Kennington (1986) e Gavish e Srikanth (1986). Ambos usam a relaxação lagrangeana para relaxar as restrições de grau do problema.

Uma das primeiras heurísticas propostas para solução do  $m$ -PCV foi o trabalho de Russel (1977) que é uma extensão da versão de Lin e Kernighan (1973). Fogel (1990) propõe um método utilizando programação evolucionária. Esta abordagem considera dois caixeiros e uma função objetivo que minimiza a diferença entre os tamanhos das rotas de cada caixeiro. Entre as metaheurísticas também surgiram diversos trabalhos com Algoritmos Genéticos (Zhang *et al*, 1999; Tang *et al*, 2000; Yu *et al*, 2002), Redes Neurais (Wacholder *et al*, 1989; Hsu *et al*, 1991), Busca Tabu (Ryan *et al*, 1998) e *Simulated Annealing* (Song *et al*, 2003).

O intuito de transformar um problema  $m$ -PCV no PCV é utilizar os métodos de resolução do PCV. Gorenstein (1970) propôs que um PCV com  $m$  caixeiros pode ser resolvido usando um PCV aumentado com  $m-1$  cidades origem adicionais, onde um custo infinito é designado para as distâncias de origem para origem, proibindo assim esse tipo de *link*. A solução de custo mínimo para o PCV e para o  $m$ -PCV será a mesma. Svestka e Huckfeldt (1973) sugerem uma transformação onde a matriz de distância original é aumentada com  $m-1$  novas linhas e colunas de tal forma que cada nova linha e coluna seja uma duplicação da primeira linha e coluna. Então

o algoritmo resolveria o problema de designação através das restrições  $\sum_{j=2}^n x_{1j} = 1, j = 2, \dots, n$  e

$\sum_{j=1}^n x_{1j} = 1, i = 2, \dots, n$  e verifica se alguma restrição de SEC foi violada. Caso uma violação tenha

ocorrido, a matriz de distância é modificada para introduzir a restrição violada no problema e o problema de designação resultante é resolvido novamente. Este processo continua até que todas as restrições sejam satisfeitas.

Outros métodos são abordados em Bektas (2006) e Svestka e Huckfeldt (1973).

## 4 REVISÃO DO ESTADO DA ARTE

O objetivo deste capítulo é fazer uma revisão do estado da arte na literatura sobre os métodos de resolução do PRV. Para facilitar a leitura os métodos foram divididos em três grandes classes: métodos exatos, heurísticas clássicas e metaheurísticas.

### 4.1 ESTADO DA ARTE SOBRE OS MÉTODOS PARA RESOLUÇÃO DO PRV

O PRV é um problema do tipo *NP-Hard* (Lenstra e Rinooy Kan, 1981) sendo um problema muito difícil de ser resolvido por métodos exatos, em função do alto custo computacional associado. Segundo Kytöjoki *et al* (2005) e Cordeau *et al* (2002) os métodos exatos têm sido usados para resolver problemas que envolvem em torno de 50 clientes. Segundo Laporte e Norbert (1987), os métodos exatos propostos podem ser classificados em três tipos: métodos de árvore de busca direta, programação dinâmica e programação linear inteira. Neste trabalho será feita uma breve apresentação de alguns métodos exatos propostos para a resolução do PRV. Para uma revisão mais detalhada é sugerida a leitura de Laporte e Nobert (1987) e Laporte (1992a). Devido ao alto custo computacional gerado pela execução dos métodos exatos, a maioria dos métodos propostos para a resolução do PRV baseia-se em heurísticas. Osman e Laporte (1996) definem heurística como sendo uma técnica que procura boas soluções (quase ótimas) com um custo computacional razoável, entretanto sem ser capaz de garantir a otimalidade. Segundo Laporte *et al* (2000) as heurísticas executam uma exploração relativamente limitada do espaço de busca e geralmente produzem soluções de boa qualidade. Ao longo do tempo, diversas heurísticas têm sido propostas para a resolução do PRV. Laporte *et al* (2000) classifica as heurísticas em duas classes principais: heurísticas clássicas e Metaheurísticas. Alguns trabalhos (Marinakis e Migdalas, 2002; Laporte e Semet, 1998) classificam as heurísticas clássicas em três tipos: métodos construtores, métodos de duas fases e métodos de melhoramento. Também é comum encontrar trabalhos (Gendreau, Hertz e Laporte, 1994; Rego, 1998; Breedam, 1995) aonde consta também um quarto método, os métodos de otimização incompleta. Os



métodos construtores criam rotas através da adição sucessiva de clientes ainda não atendidos nas rotas em formação de acordo com algum critério estabelecido. Entre os métodos construtores encontram-se as heurísticas clássicas de Clarke e Wright (1964), Mole e Jamenson (1976) e diversas heurísticas para a resolução do problema do Caixeiro Viajante. Os métodos de duas fases são assim chamados porque o processo de criar rotas é dividido em duas etapas: em uma etapa de criação de *clusters* e outra etapa de geração das rotas. Os métodos de duas fases podem ser chamados de *cluster-first, route-second* ou métodos *route-first, cluster-second*. No primeiro caso, inicialmente agrupam-se os clientes e depois, geram-se as rotas, já no segundo caso uma rota é construída com todos os clientes e depois a rota é segmentada formando grupos de clientes. Nesta classe encontra-se a heurística de Gillet e Miller (1974) e a heurística de Fisher e Jaikumar (1981). Os métodos de melhoramento buscam melhorar uma solução inicial através da troca ou exportação de clientes entre as rotas. Este tipo de método pode ser usado para melhoramento de uma única rota (*intra-route*) ou de várias rotas simultaneamente (*inter-route*). Segundo Laporte e Semet (1998) e Laporte *et al* (2000) a maioria dos métodos de melhoramento *intra-route* podem ser descritos em termos dos mecanismos  $\lambda$ -opt de Lin (1965). Com relação aos trabalhos de melhoramento *inter-route*, Breedam (1995) faz uma análise e considera quatro tipos possíveis de operações a serem feitas: *String Cross*, *String Exchange*, *String Relocation* e *String Mix*. Outras referências deste tipo de heurística podem ser encontradas no trabalho de Thompson e Psaraftis (1993). As heurísticas de melhoramento são também conhecidas como Heurísticas de Busca Local (Marinakis e Migdalas, 2002; Thangiah e Petrovic, 1997). As heurísticas clássicas vem sendo utilizadas como método para resolução do PRV durante os últimos 40 anos, entretanto a partir das últimas duas décadas diversas metaheurísticas começaram a ser utilizadas como solução para este problema. O desenvolvimento das metaheurísticas foi iniciado a partir dos anos 90 e desde então elas vêm sendo aplicadas com sucesso na resolução de problemas de otimização combinatória. Osman e Laporte (1996) definem metaheurística como sendo um processo iterativo que guia uma heurística subordinada através da combinação inteligente de diferentes conceitos provenientes de fenômenos naturais e exploração do espaço de busca, usando estratégias de

aprendizado para estruturar as informações com o objetivo de se obter soluções eficientes perto da solução ótima. A grande vantagem das metaheurísticas é que elas aceitam deixar um ótimo local, através da execução de movimentos que causem uma deterioração da função objetivo (Breedam, 2001), a fim de que possam continuar buscando um ótimo global. Diferente das heurísticas clássicas, nas metaheurísticas, a ênfase está em executar uma exploração profunda nas regiões mais promissoras do espaço de busca utilizando sofisticadas regras de geração de vizinhança, estruturas de memória e recombinação de soluções (Laporte *et al*, 2000). Diversas metaheurísticas têm sido propostas para a resolução do PRV. Entre as metaheurísticas que Gendreau *et al* (1999) destaca como aplicáveis na resolução do PRV estão: 1) *Simulated Annealing* (Osman, 1993; Breedam, 1995); 2) *Deterministic Annealing* (Dueck, 1993; Dueck e Scheurer, 1990); 3) Busca Tabu (Osman, 1993; Gendreau, Hertz e Laporte, 1994; Taillard, 1993; Xu e Kelly, 1996; Rego e Roucairol, 1996; Barbarasoglu e Ozgur, 1999; Rochat e Taillard, 1995; Toth e Vigo, 2003; Cordeau *et al*, 2002); 4) Algoritmos Genéticos (Breedam, 1995); 5) Colônia de Formigas (Kawamura *et al*, 1998) e 6) Redes Neurais (Matsuyama, 1991). Breedam (2001), Baker e Ayechev (2003) destacam *Simulated Annealing* e Busca Tabu como as metaheurísticas mais eficientes na resolução do PRV e entre elas Cordeau *et al* (2002), Toth e Vigo (2003), Prins (2004), Xu e Kelly (1996) afirmam que a Busca Tabu é a Metaheurística de maior destaque nessa área. Para uma revisão mais aprofundada sobre Metaheurísticas recomenda-se a leitura de Osman e Laporte (1996), Rayward-Smith *et al* (1996) e Aarts e Lenstra (Aarts e Lenstra, 1997). Os trabalhos de Tarantilis *et al* (2005), Gendreau, Laporte e Potvin (1999) e Cordeau *et al* (2002) são sugeridos para uma revisão sobre as metaheurísticas aplicadas ao PRV.

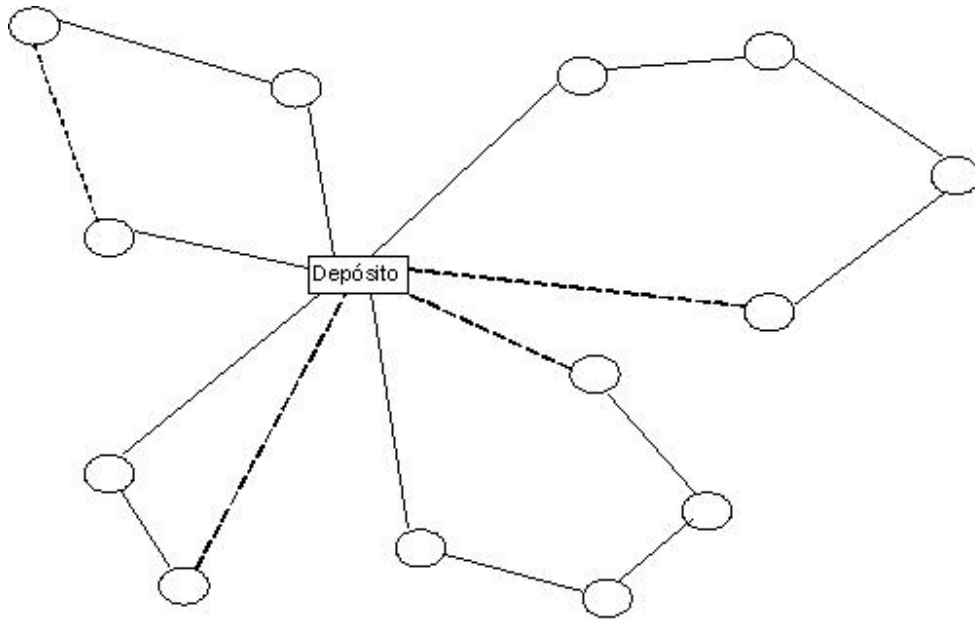
#### **4.1.1 Métodos Exatos para Resolução do PRV**

Segundo Laporte e Norbert (1987), os métodos exatos propostos podem ser classificados em três grandes grupos: métodos de árvore de busca direta, programação dinâmica e programação linear inteira. É comum os trabalhos da literatura (Laporte, 1992a; Rego e Roucairol, 1996; Xu e Kelly, 1996; Cordeau *et al*, 2002; Tarantilis, Ioannou e Prastacos, 2005)

afirmarem que os métodos exatos são indicados apenas para pequenas e médias instâncias do problema, pois tendem a tornar-se ineficientes à medida que a quantidade de clientes aumenta. Contudo, na literatura sobre o PRV é possível encontrar diversos trabalhos que utilizam esses métodos. Laporte e Nobert (1987), Laporte (1992a) apresentam um *survey* com diversos métodos exatos para resolução do PRV baseados principalmente em algoritmos de *branch-and-bound*, relaxações diversas, programação dinâmica e algoritmos para formulações de fluxo de veículos. Christofides e Eilon (1969) apresentam um dos primeiros algoritmos de *branch-and-bound* para o PRV onde o problema é formulado como PCV e tem seus limites inferiores calculados a partir de árvores de expansão mínima. Christofides, Mingozzi e Toth (1981), Fisher (1994), Hadjicostantinou *et al* (1995) apresentam algoritmos de *branch-and-bound* utilizando as seguintes relaxações respectivamente, *k-degree center tree* e q-rotas, *k-trees* e particionamento de conjuntos. O trabalho de Ralphs *et al* (2001) apresenta um algoritmo de *branch-and-cut* construído a partir de uma metodologia de separação baseada em decomposição para as restrições de capacidade. Toth e Vigo (2002) descrevem detalhadamente o PRV capacitado no caso simétrico e assimétrico. No caso assimétrico, o problema é modelado através de formulações de fluxo de veículos com limites inferiores calculados a partir de modelos de designação, arborescências e fluxo de custo mínimo, e no caso simétrico, é apresentada uma formulação de fluxo de veículos com duplo índice e relaxações baseadas em árvore de expansão mínima, *b-matching* e no problema de partição de conjuntos. Martinhon, Lucena e Maculan (2004) apresentam um algoritmo de *relax and cut* para resolver o PRV formulado como uma relaxação da *k-tree*, estendendo o trabalho de Fisher (Fisher, 1994).

Segundo Hillier e Lieberman (1988) métodos de *branch-and-bound* são, basicamente, procedimentos de enumeração. Os autores explicam o funcionamento geral de um algoritmo de *branch-and-bound* da seguinte maneira: supondo que exista uma função objetivo que se deseje minimizar e um limite superior para esta função esteja disponível, o primeiro passo é dividir o conjunto de todas as soluções viáveis em diversos subconjuntos e para cada um deles obter um limite inferior. Os subconjuntos cujos limites ultrapassam o limite superior são excluídos. Dos

subconjuntos restantes, um deles é escolhido e novamente dividido em diversos subconjuntos. Este procedimento continua até que seja encontrada uma solução viável tal que o valor correspondente da função objetivo não seja maior que o limite inferior para qualquer subconjunto. Garfinkel (1979) faz uma boa apresentação dos conceitos relacionados a esta técnica. A seguir detalhes sobre alguns trabalhos que utilizam essa técnica como solução do PRV. O algoritmo Christofides e Eilon (1969) usa relaxação baseada em árvores de expansão mínima para calcular os limites inferiores nos vértices e antes como condição para executar o corte testa se qualquer uma das 3 condições a seguir é verdadeira: (1) a carga total do veículo excede a capacidade, (2) a distância total do veículo excede o limite ou (3) a capacidade restante dos veículos não é capaz de atender as cidades ainda não visitadas. No trabalho de Christofides, Mingozzi e Toth (1981) foi desenvolvido um algoritmo de *branch-and-bound* baseado nas técnicas de árvore de expansão mínima e q-rotas associados ao uso de relaxação lagrangeana para gerar os limites inferiores. Para calcular os limites inferiores baseados em árvore de expansão mínima, os autores modelam o PRV através de sua generalização, o *m*-PCV, sugerindo que uma solução para o *m*-PCV seria um limite inferior para o PRV. Os autores denominam de “*k-degree center tree*” (*k*-DCT) uma solução para o *m*-PCV. Na figura 3 uma solução com  $M = 4$  e  $k = 5$  é ilustrada. A figura mostra a partição de todas as arestas em subconjuntos. As arestas representadas por linhas cheias (\_\_\_\_) fazem parte do conjunto *k-degree center tree*. As arestas representadas por linhas tracejadas (-----) são as arestas adjacentes ao depósito e as arestas representadas por linhas pontilhadas (.....) são as não adjacentes ao depósito.



**FIGURA 3** - Partição em subconjuntos. \_\_\_\_\_  $k$ -degree center tree( $k=5$ ),----- $y$  arestas adjacentes ao depósito ( $y=3$ ), ..... $M-y = 1$  aresta não adjacentes ao depósito. Fonte: Laporte (1992b)

Segundo Christofides, Mingozzi e Toth (1981), se em qualquer conjunto  $S_0$  de  $y \leq M$  arcos adjacentes ao vértice  $x_0$  (depósito) e qualquer  $S_1$  de  $M-y$  arcos não adjacentes a  $x_0$  forem removidos - um de cada rota - o resultado irá formar uma  $k$ -DCT com  $k = 2M - y$ . Desta maneira, os arcos que formam a  $k$ -DCT são separados em 3 grupos associados as seguintes variáveis binárias:  $\xi_l = 1$  se o arco  $l$  pertence a  $k$ -DCT, e 0 caso contrário;  $\xi_l^0 = 1$  se o arco  $l$  pertence ao conjunto  $S_0$  e 0 caso contrário e  $\xi_l^1 = 1$  se o arco  $l$  está presente no conjunto  $S_1$  e 0 caso contrário. Assim sendo, o  $m$ -PCV é formulado como:

$$\text{Minimizar } z = \sum_{l=1}^m c_l (\xi_l + \xi_l^0 + \xi_l^1), \quad (4.1)$$

sujeito a:

$$\sum_{l \in (S, \hat{S})} \xi_l \geq 1, \forall S_i \subset X; (S_i \geq 1); \quad (4.2)$$

$$\sum_{l \in A_0} \xi_l = 2m - y; \quad (4.3)$$

$$\sum_{l=1}^m \xi_l = N; \quad (4.4)$$

$$\sum_{l \in A_0} \xi_l^0 = y; \quad (4.5)$$

$$\sum_{l \in A - A_0} \xi_l^1 = M - y; \quad (4.6)$$

$$\sum_{l \in A_i} (\xi_l + \xi_l^0 + \xi_l^1) = 2, \quad i = 1, \dots, N; \quad (4.7)$$

$$\xi_l \in \{0,1\}; \quad (4.8)$$

$$\xi_l^0 \in \{0,1\}; \quad (4.9)$$

$$\xi_l^1 \in \{0,1\}; \quad (4.10)$$

$$Y \text{ arbitrário}; \quad (4.11)$$

onde  $m = |A|$  é o número total de arcos;  $(S, \hat{S})$  é o conjunto de todos os arcos com um terminal no  $S_i$  e outro no complemento do  $\hat{S}_i$ ;  $A_i$  é o conjunto de todos os arcos incidentes em  $x_i$ . A equação (4.1) representa a função objetivo, que minimiza a soma dos arcos dos três conjuntos. A equação (4.2) impõe a conectividade do k-DCT. A equação (4.3) impõe o grau no vértice depósito. As

equações (4.4), (4.5) e (4.6) especificam o número de arcos requeridos. A equação (4.7) impõe o grau de todos os vértices que não são o depósito. Sendo definido  $\lambda_i$ ,  $i = 1, \dots, N$ , como uma penalidade não-negativa associada à restrição (4.7), a relaxação lagrangeana dessas restrições produz 3 problemas P, P<sub>0</sub> e P<sub>1</sub> para um dado valor de y, com o objetivo geral:

$$V(\lambda, y) = \sum_{l=1}^m c'_l w_l - 2 \sum_{i=1}^N \lambda_i, \quad (4.12)$$

onde  $c'_l = c_l + \lambda_{il} + \lambda_{jl}$ , il e jl são dois vértices do arco l,  $\lambda_0 = 0$  e  $w_l$  é escrito para  $\xi_l$ ,  $\xi_l^0$  e  $\xi_l^1$ .

Então para um dado valor de y e  $\lambda$ ,  $V(\lambda, y)$  é o valor ótimo para o problema definido pelas equações (4.12), (4.2), (4.3), (4.4) e (4.8).  $V^0(\lambda, y)$  é a solução ótima para o problema P<sub>0</sub> definido por (4.12), (4.5) e (4.9) e  $V^1(\lambda, y)$  é a solução ótima para o problema P<sub>1</sub> definido por (4.12), (4.6) e (4.10). Um limite inferior para o m-PCV é:

$$H = \underset{M1 \leq y \leq M}{Max} \{ \underset{\lambda \geq 0}{Max} [ V(\lambda, y) + V^0(\lambda, y) + V^1(\lambda, y) ] \}.$$

Os autores mostram que os subproblemas P, P<sub>0</sub> e P<sub>1</sub> podem ser resolvidos em tempo polinomial. Para maiores detalhes sobre esta abordagem é sugerida a leitura do trabalho original.

Laporte, Mercure e Nobert (1986) resolvem o problema através de um algoritmo de *branch-and-bound* em que os subproblemas são formulados como problemas de designação. Neste trabalho é explorada a relação entre o PRV, m-PCV e o PCV. O algoritmo é desenvolvido através da transformação do m-PCV no PCV clássico através da adição de variáveis depósito artificiais  $m_u$  e geração de matriz de custo estendida. Sendo  $n' = n + m_u - 1$  a nova quantidade de vértices,  $V' = \{1, \dots, n'\}$  o novo conjunto de vértices e  $A' = A \cup \{(i, j): i, j \in V', i \neq j, i \text{ ou } j \in V' \setminus V\}$  o novo conjunto de arcos e PRV pode ser formulado como (Laporte, 1992a):

$$\text{Minimizar } \sum_{i \neq j} c'_{ij} x_{ij}; \quad (4.13)$$

sujeito a:

$$\sum_{j=1}^{n'} x_{ij} = 1 (i = 1, \dots, n'); \quad (4.14)$$

$$\sum_{j=1}^{n'} x_{ij} = 1 (i = 1, \dots, n'); \quad (4.15)$$

$$\sum_{i,j \in S} x_{ij} \leq |S| - v(S) \quad (S \subset V \setminus \{1\}; |S| \geq 2); \quad (4.16)$$

$$x_{ij} \in \{0,1\} \quad (i,j = 1, \dots, n'; i \neq j); \quad (4.17)$$

onde as equações (4.13),(4.14),(4.15) e (4.17) definem o problema de designação modificado e a equação (4.16) representa a eliminação de *subtour*.  $v(S)$  é um limite inferior do número de veículos necessários para visitar todos os vértices de  $S$  na solução ótima. No contexto do PRV

$$\text{Capacitado, } v(S) = \left\lceil \frac{\sum_{i \in S} d_i}{D} \right\rceil.$$

O algoritmo de resolução é semelhante ao algoritmo proposto em Laporte (1992b) para o PCV, entretanto com algumas diferenças na definição das *subtours* inválidas. Para interpretar o resultado do algoritmo como solução para o PRV, as seguintes regras são necessárias:

- (i) se  $i \in V \setminus \{1\}$  e  $j \in V \setminus V$ , substitua  $(i,j)$  por  $(i,1)$ ;
- (ii) se  $i \in V' \setminus V$  e  $j \in V \setminus \{1\}$ , substitua  $(i,j)$  por  $(1,j)$ ;
- (iii) se  $i,j \in V' \setminus V$ , delete  $(i,j)$ .

Hadjcostantinou *et al* (1996) propuseram um algoritmo de *branch-and-bound* baseado na relaxação do PRV para o problema de particionamento de conjuntos. Segundo Toth e Vigo



(2002) e Laporte (1992a) a formulação como problema de partição de conjuntos para o PRV foi proposta primeiramente por Balinsky e Quandt (1964). A redução para o problema de partição de conjuntos é abordada em Czeck (1999) como: Definindo-se o valor de  $k$  como a quantidade máxima de clientes que podem ser visitados em uma rota e tornando  $k$  uma restrição do problema, o PRV se reduz ao problema de particionamento de conjuntos. Sendo  $N = \{1,2,\dots,n\}$  um conjunto de clientes,  $P = \{P_1, P_2, \dots, P_s\}$ ,  $s = \binom{n}{1} + \binom{n}{2} + \dots + \binom{n}{k}$ , o conjunto de todos os subconjuntos de  $N$  de tamanho máximo  $= k$ ,  $P_i \subseteq N$  e  $|P_i| \leq k$ ,  $i \in M$ , onde  $M = \{1,2,\dots,s\}$ . Cada  $P_i$  representa uma *tour* da solução para o PRV e  $c_i$  o custo mínimo da *tour*  $P_i$ . A resolução do PRV consiste em resolver o problema de partição de conjuntos e encontrar a coleção  $\{P_i\}$ ,  $i \in M$ , de custo mínimo total tal que cada cliente seja coberto pelos subconjuntos da coleção exatamente uma única vez.

Laporte (1992a) apresenta a seguinte formulação baseada no problema de partição de conjuntos: Seja  $J$ , o conjunto de todas as rotas possíveis  $j$  e  $a_{ij}$  uma variável binária igual a 1 se o vértice  $i > 1$  aparece na rota  $j$ . Seja  $C_j^*$  o custo ótimo da rota  $j$  e  $x_j$  uma variável binária igual a 1 se a rota  $j$  é utilizada na solução ótima. A formulação apresentada é:

$$\text{Minimize } \sum_{j \in J} C_j^* x_j; \quad (4.18)$$

sujeito a:

$$\sum_{j \in J} a_{ij} x_j = 1, (i \in V \setminus \{1\}); \quad (4.19)$$

$$x_j \in \{0,1\}, (j \in J); \quad (4.20)$$

onde a equação (4.18) é relativa à função objetivo que deseja minimizar o custo das rotas e a equação (4.19) impõe que cada cliente deva ser atendido por uma única rota.

Laporte e Nobert (1987) apontam como as duas maiores dificuldades deste tipo de abordagem a quantidade de variáveis binárias e a dificuldade de calcular  $C_j^*$ . No caso do PRV capacitado, cada rota  $j$  corresponde a um conjunto de cidades  $S_j$  satisfazendo  $\sum_{i \in S_j} d_i \leq D$ . Para calcular o valor de  $C_j^*$ , resolve-se o PCV em  $S_j$ .

#### 4.1.2 Heurísticas Clássicas para a resolução do PRV

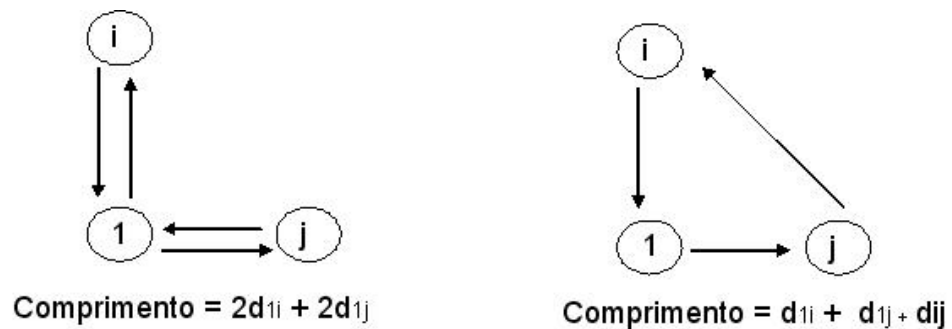
A seguir, são apresentadas algumas heurísticas clássicas (Laporte *et al*, 2000) utilizadas na resolução do PRV:

##### 4.1.2.1 Algoritmo de Clarke e Wright

Esta, talvez, seja a heurística mais conhecida para resolução do PRV (Thangiah e Petrovic, 1997; Laporte *et al*, 2000). Este método, também conhecido como o método das economias, inicia assumindo que um único veículo serve um único cliente. Conseqüentemente, para um conjunto de  $N$  clientes, o método assume que  $N$  veículos serão requeridos. O método então calcula a economia  $S_{ij}$ , em distância, que pode ser obtida unindo os clientes  $i$  e  $j$  e atendendo a eles com um único veículo. A economia  $S_{ij}$  é calculada através da fórmula:

$$S_{ij} = S_{ji} = d_{i1} + d_{1j} - d_{ji}.$$

A seguir, a figura 4 representa a economia obtida através da união dos clientes  $i$  e  $j$  que passam a ser atendidos por um único veículo:



**FIGURA 4** - Representação da economia atingida com a utilização de um mesmo veículo para atender os vértices i e j. Fonte: Goldbarg e Luna (1998).

Assim sendo, a economia é obtida pela redução do número de veículos requeridos para atender os clientes i e j por um aumento na distância a ser viajada pelo veículo que atende os clientes i e j. As economias calculadas são ordenadas decrescentemente e os clientes i e j que têm as maiores economias são unidos enquanto a capacidade de restrição não é ultrapassada. O método une clientes sem violar as restrições de capacidade e pára quando nenhuma união a mais pode ser feita.

O algoritmo de Clark e Wright é simples de implementar permitindo que diversas restrições sejam incorporadas facilmente (Goldbarg e Luna, 2000), além disso, produz rotas com veículos que têm alta taxa de utilização em termos de capacidade e distância (Thangiah e Petrovic, 1997).

#### 4.1.2.2 Algoritmo de Mole e Jamenson

É um procedimento seqüencial de inserção (Christofides, Mingozzi e Toth, 1979) no qual para um dado valor de 2 parâmetros  $\lambda$  e  $\mu$ , os dois critérios a seguir são usados para expandir a rota em construção:

$$e(i,l,j) = c_{il} + c_{lj} - \mu c_{ij},$$

$$\sigma(i,l,j) = \lambda c_{0l} - e(i,l,j).$$

O algoritmo procede da seguinte maneira:

Passo 1: Para cada cliente ainda não alocado a nenhuma rota  $x_l$ , calcule a inserção viável na rota emergente R como

$$e(i,l,j_l) = \min[e(r,l,s)],$$

para todos clientes adjacentes  $x_r, x_s \in R$ , onde  $x_{il}$  e  $x_{jl}$  são os clientes entre os quais  $x_l$  resulta na melhor inserção

Passo 2: O melhor cliente  $x_{l^*}$  a ser inserido na rota é calculado como aquele para o qual

$$\sigma(i_{l^*},l^*,j_{l^*}) = \text{Max}[\sigma(i_l,l,j_l)],$$

onde  $x_l$  é não alocado e possível.

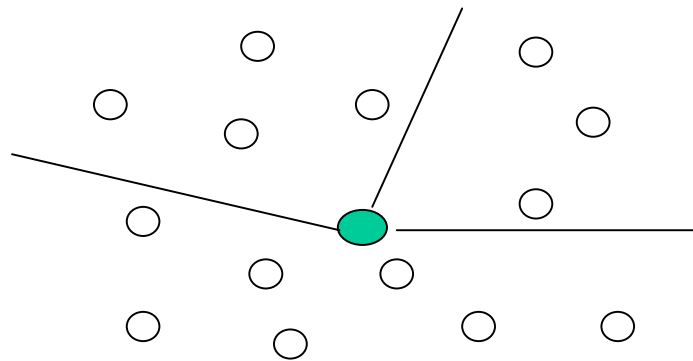
Passo3: Insira  $x_{l^*}$  na rota R entre  $x_{il^*}$  e  $x_{jl^*}$ .

Passo 4: Otimize a rota R usando métodos r-otimos.

Passo 5: Retorne ao passo 1 para iniciar a criação de uma nova rota R até que todos os clientes tenham sido alocados em rotas

#### 4.1.2.3 Algoritmo de Gillet e Miller

Também conhecida como o algoritmo das varreduras (Thangiah e Petrovic, 1997). Esta heurística procura obter a solução em duas etapas distintas: primeiro agrupar os pontos de demanda segundo algum critério de proximidade formando assim *clusters*, depois solucionar cada *cluster* independentemente criando uma rota para estes clientes (Goldberg e Luna, 2000). A figura 5 abaixo ilustra uma possível construção de clusters:



**FIGURA 5** - Ilustração dos *clusters* gerados pelo método das varreduras. Fonte: Bodin, Golden e Assad (1983).

Esta heurística usa além das distâncias entre os clientes, os ângulos das coordenadas polares para construir as rotas. Os clientes são organizados numa ordem crescente (ou decrescente) dos seus ângulos em coordenadas polares. O algoritmo inicialmente forma os *clusters* unindo os clientes sem violar a restrição da capacidade. Num segundo momento são criadas rotas para cada um destes clusters. A criação destas rotas pode ser feita através de algoritmos para resolução do PCV ou usando-se algoritmos exatos (Thangiah e Petrovic, 1997)

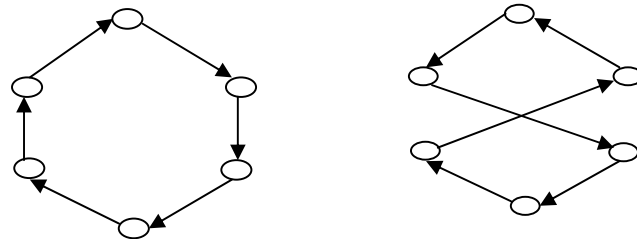
#### 4.1.2.4 Heurísticas de Busca Local ou de Melhoramento

Segundo Toth e Vigo (2002) os algoritmos de busca local para o PRV geralmente usam vizinhanças simples baseadas na troca de arcos ou movimentos entre clientes. Por exemplo, dada uma solução atual  $s$ , uma vizinhança baseada na  $k$ -troca é obtida através da remoção de  $k$  arcos usados em  $s$  e sua substituição com outros  $k$  arcos que definem uma nova solução.

##### 4.1.2.4.1 Heurísticas de Melhoramento *Intra-Route*

Este tipo de heurística é aplicada às rotas individualmente. Estes métodos fazem uma mudança na ordem em que os clientes são visitados dentro de cada rota (Thangiah e Petrovic, 1997), na tentativa de diminuir a distância ou tempo da rota (Breedam, 1995). Segundo Laporte

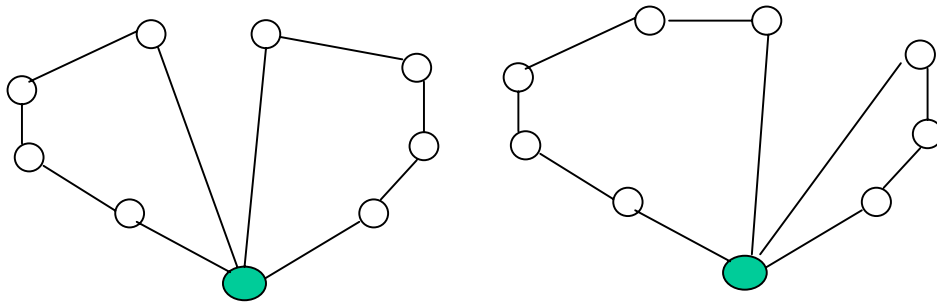
*et al* (2000) e Laporte e Semet (1998) a maioria dos métodos existentes podem ser descritos em termos dos mecanismos  $\lambda$ -opt de Lin (1965). Nesse caso,  $\lambda$  arestas são removidas da *tour* e as  $\lambda$  restantes são reconectadas de todas as possíveis maneiras. Se algumas destas novas conexões proporcionar um custo menor, ou seja, se melhorar a rota, este movimento é executado. Para Thangiah e Petrovic (1997) os métodos mais comuns são os métodos 2-opt e 3-opt. O método 2-opt remove duas arestas de uma rota inicial e então cria duas novas arestas diferentes ligando os clientes que ficaram desconectados. O método 3-opt segue o mesmo procedimento, entretanto 3 vértices são removidos. A figura 6 ilustra um possível movimento realizado na rota, alterando a seqüência em que os clientes são visitados:



**FIGURA 6-** Ilustração do método de otimização dentro da rota. Fonte: Thangiah e Petrovic (1997).

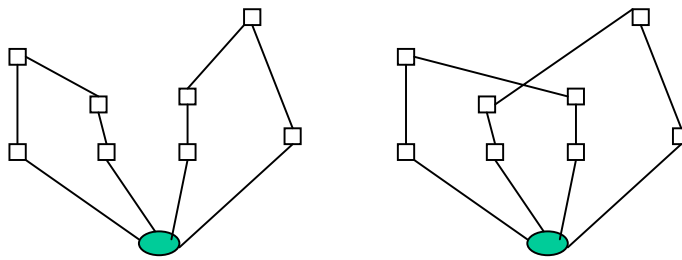
#### 4.1.2.4.2 Heurísticas de Melhoria *Inter-Route*

Este tipo de heurística considera possíveis trocas ou realocações de clientes entre cada duas rotas (Breedam, 1995), sempre respeitando as restrições associadas. A figura 7 ilustra um possível movimento de melhoria entre duas rotas.

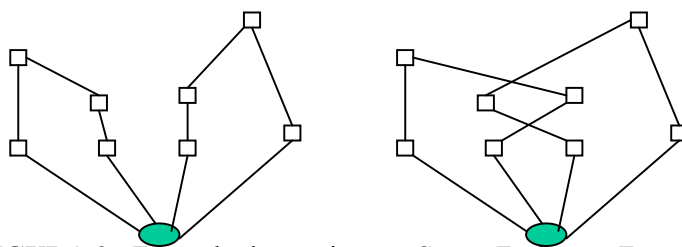


**FIGURA 7** - Ilustração do método de otimização entre rotas. Fonte: Thangiah e Petrovic (1997).

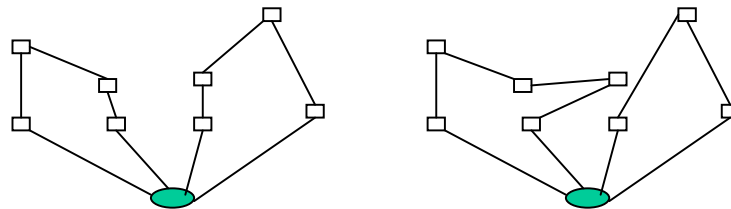
Thompson e Parafitis (1993) apresentam um esquema “*b-cyclic, k-transfer*” no qual uma permutação circular de  $b$  rotas é considerada e  $k$  clientes de cada rota são transferidos para a próxima rota do ciclo da permutação (Laporte *et al*, 2000). Breedam (1995) classifica os movimentos como *String Cross*, *String Exchange*, *String Relocation* e *String Mix* e segundo Laporte *et al* (2000) e Laporte e Semet (1998) estes movimentos podem ser vistos como casos especiais da troca *2-cyclic*. As figuras 8, 9 e 10 a seguir ilustram a classificação de Breedam.



**FIGURA 8** - Exemplo do movimento *String Cross*. Fonte: Breedam (2001).



**FIGURA 9** - Exemplo do movimento *String Exchange*. Fonte: Breedam (2001).



**FIGURA 10-** Exemplo do movimento *String Relocation*. Fonte: Breedam (2001).

#### 4.1.2.5 Heurísticas Clássicas para resolução do PCV

Conforme foi visto no capítulo 3, a associação do PCV ao PRV é bastante natural o que implica em possibilitar o uso de heurísticas clássicas desenvolvidas para o PCV na resolução do PRV com algumas adaptações, no sentido de admitir as restrições adotadas no PRV, como capacidade dos veículos ou tempo de duração da rota. A seguir são apresentadas algumas heurísticas clássicas do PCV que podem ser usadas como solução no PRV.

##### 4.1.2.5.1 Método do Vizinho mais Próximo

O trabalho de Tyagi (1968) apresenta o método do Vizinho mais Próximo para a resolução do PRV. Este procedimento deve, a partir de um dado vértice, visitar o vértice mais próximo que ainda não foi visitado e retornar ao vértice inicial quando todos os outros tiverem sido visitados. Segundo Cook *et al* (1998), a vantagem deste método é que ele é fácil de ser implementado, rápido, e geralmente produz *tours* de qualidade razoável.

##### 4.1.2.5.2 Procedimentos de Inserções

Este método inicia com um *tour* unindo  $k$  vértices, então adiciona os vértices restantes um a um conforme a política escolhida, sendo que esta pode assumir diversas variações, como por exemplo:

- Inserção do mais Distante: neste caso, inicia-se uma *tour* com dois vértices que estão no final de uma aresta de alto custo. Para cada vértice não inserido  $v$ , é



calculado o custo mínimo entre  $v$  e qualquer vértice na *tour* construído. Então é escolhido como o próximo a ser inserido aquele para o qual o custo for máximo (Cook *et al*, 1998; Reinelt, 1994).

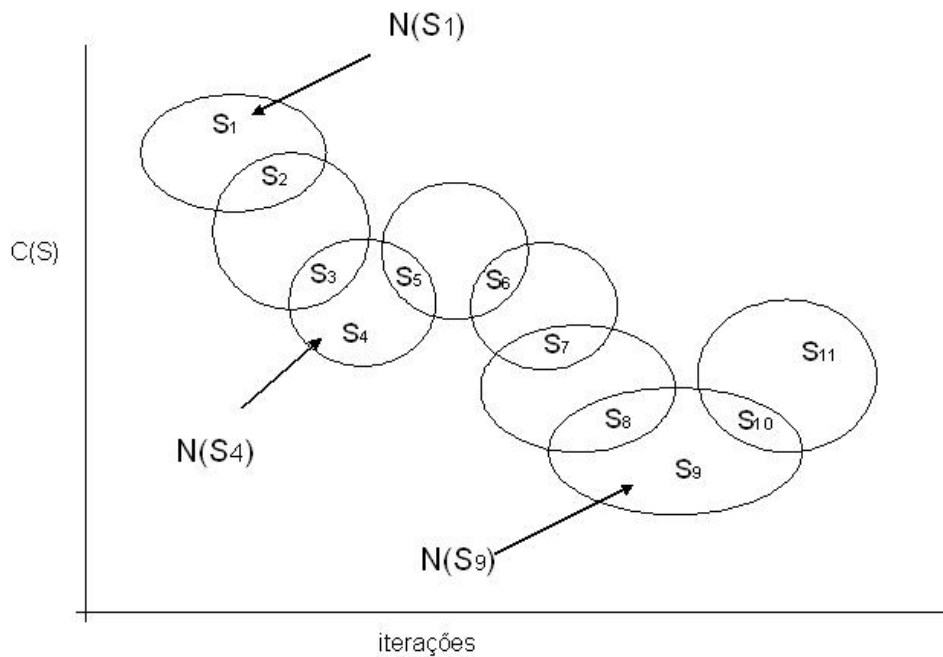
- Inserção do mais Próximo: inicia-se o subgrafo com apenas um vértice, encontra-se um outro para formar a *subtour*  $i$ - $k$ - $i$ , cujo custo  $c_{ik}$  seja mínimo. Dada esta *subtour* seleciona-se o próximo vértice como sendo o mais próximo a qualquer vértice na *subtour* e o insere entre  $i$ - $k$  (Cook *et al*, 1998).
- Inserção do mais Barato: semelhante à inserção do mais próximo, com exceção de que o próximo vértice a ser inserido não precisa ser o mais próximo a qualquer um da *subtour* já existente (Cook *et al*, 1998; Reinelt, 1994).
- Inserção arbitrária: também inicia com um subgrafo, porém, o próximo vértice a ser inserido é escolhido aleatoriamente (Bodin, Golden, Assad e Ball, 1983).

### 4.1.3 Metaheurísticas

Nos últimos anos diversos pesquisadores vêm utilizando as metaheurísticas como solução do PRV. A literatura (Cordeau *et al*, 2002; Toth e Vigo, 2003; Prins, 2004; Xu e Kelly, 1996) indica que a Busca Tabu oferece melhores resultados do que outras metaheurísticas.

A Busca Tabu foi proposta por Glover (1986) e teve seus conceitos aprofundados em Glover e Laguna (1997). Consiste numa rotina iterativa para construir vizinhanças com ênfase na proibição do bloqueio em um ótimo local. Segundo Osman e Laporte (1996) o emprego de estratégias para modificar a vizinhança de uma solução durante o processo de busca permite que se ultrapasse os limites de uma heurística de busca local comum. A Busca Tabu utiliza estruturas auxiliares de memória de longo e curto prazo, também chamadas de memória baseada em frequência e recência respectivamente, para determinar as novas vizinhanças.

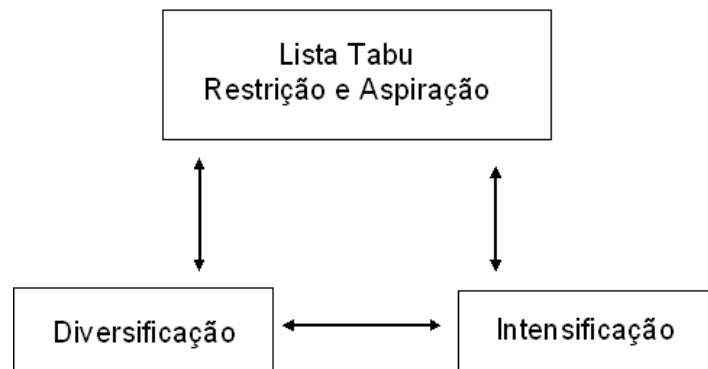
As estruturas de memória de longo prazo geralmente são utilizadas para fornecer um tipo de informação que complementa as informações da memória de curto prazo. Glover e Laguna (1997) apresentam o conceito de frequência como taxas expressas através de duas medidas: medidas de transição (número de iterações em que um atributo muda de status) e medidas de residência (número de iterações em que um atributo mantém um status). Geralmente, o uso da memória baseada em frequência é associado à criação de penalidades ou incentivos para modificar a avaliação dos movimentos e gerar uma diversificação das soluções. Com relação às estruturas de memória de curto prazo, a mais utilizada é a Lista Tabu. A Lista Tabu é uma estrutura que armazena os atributos definidos como sendo proibidos evitando que eles estejam presentes nas próximas  $T$  iterações, sendo  $T$  o tamanho definido para a Lista Tabu. O objetivo de proibir atributos recentemente utilizados é evitar ciclos em torno de ótimos locais e forçar um avanço no espaço de busca. Em geral, tamanhos pequenos da lista permitem a exploração de soluções perto dos ótimos locais, enquanto que uma lista de tamanho grande força a busca a se distanciar de ótimos locais. A figura 11 ilustra um processo de busca no qual a solução ótima ultrapassa o valor que seria obtido através de busca local comum.



**FIGURA 11** - Processo de busca continuado além dos ótimos locais. Fonte: Osman e Laporte (1996).

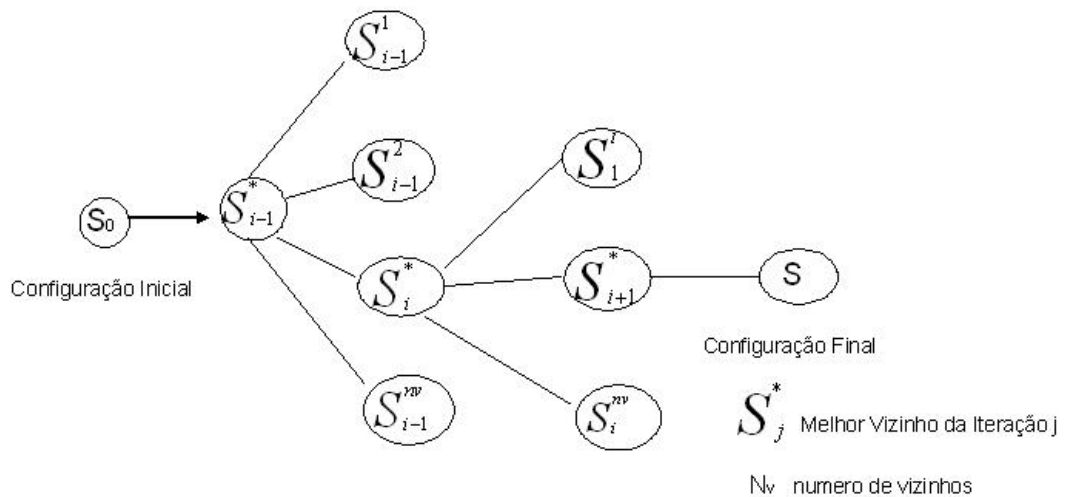
Segundo Viana (1998) a Busca Tabu tem por base três princípios: i) uso de estrutura de dados tipo fila (Lista Tabu) para manter o histórico da evolução da busca; ii) uso de mecanismos de controle para fazer um balanceamento entre aceitação ou não de uma nova solução e iii) incorporação de procedimentos que alternam as estratégias de diversificação e intensificação. Uma estratégia de intensificação foca a direção da busca para regiões historicamente consideradas boas enquanto que o uso de estratégias de diversificação força a busca examinar regiões pouco exploradas ou até mesmo desconhecidas.

A figura 12 ilustra a interação entre esses componentes.



**FIGURA 12** - Componentes da estratégia de Busca. Fonte: Viana (1998).

Segundo Glover (1990), o processo no qual o método da Busca Tabu procura a melhor solução é através de uma exploração agressiva que escolhe o melhor movimento a cada iteração independente se este movimento melhora ou não a solução atual. A figura 13 a seguir apresenta um esboço da evolução do processo de geração de  $N_v$  soluções vizinhas à atual. A notação indica  $S_i^j$ , para “j” variando de 1 a  $N_v$ , é o vizinho da ordem “j” de  $S_{i-1}$ ; e que  $S_i^*$  é a “melhor” solução entre eles, para algum  $1 \leq j \leq N_v$ . Supondo um problema de minimização, a “melhor” solução será aquela que apresentar o menor valor da função objetivo.



**FIGURA 13** – Esquema de geração de vizinhança para a Busca Tabu. Fonte: Viana (1998).

O algoritmo da Busca Tabu permite encontrar em um conjunto  $X$  de possíveis soluções viáveis uma solução  $s$  que otimize uma função objetivo  $f$ , por exemplo:

Minimizar  $f(s)$ :  $s \in X$  e  $X \in R_v$ , sendo  $R_v$  a região viável.

A vizinhança  $N(s)$  é definida como cada solução  $s$  de  $X$ . Estas vizinhanças são geradas a partir de uma solução inicial  $s_{inic}$  e a cada iteração um ótimo local é escolhido e a partir dele uma nova vizinhança será gerada. Segundo Viana (1998), no desenvolvimento da Busca Tabu, estratégias de intensificação e diversificação são alternadas através da análise dos atributos tabu. As estratégias de diversificação direcionam a busca para novas regiões, buscando atingir todo o espaço de solução possível enquanto que as estratégias de intensificação reforçam a busca na vizinhança de uma solução historicamente boa. As estratégias de intensificação são baseadas na modificação das regras de escolha para incentivar combinação de movimentos e soluções historicamente consideradas boas. Também podem determinar o retorno a regiões atrativas para aprofundar a busca naquela região. As estratégias de diversificação são baseadas na modificação de regras de escolha para trazer atributos pouco usados para as soluções. Durante o processo de

busca, para evitar ciclos em torno de um ótimo local, é implementada uma lista de movimentos proibidos, chamada Lista Tabu. Esta lista é implementada através de uma fila circular e tem um tamanho  $t_k$  definido como o número de vezes que se quer proibir a visita de uma solução. Este tamanho pode ser fixo ou variar de acordo com critérios definidos. Em geral, tamanhos pequenos permitem a exploração de soluções perto dos ótimos locais, enquanto que uma lista de tamanho grande força a busca a se distanciar de ótimos locais. Quando uma solução definida como proibida é escolhida como a melhor, ela poderá ser admitida se for aceita pelo Critério de Aspiração. O critério mais usual perdoa a solução caso o valor da sua função objetivo seja melhor do que o melhor resultado até então encontrado. Este processo de busca continua até que o Critério de Parada seja alcançado. Usualmente define-se este critério como sendo um número máximo de iterações ( $Nbmax$ ) que ocorram sem melhora na função objetivo.

A seguir é apresentado um algoritmo geral (Rodrigues e Gómez, 2000) da Busca Tabu, para o caso de minimização da função objetivo.

```

Inicialização
s := solução inicial em X;
Nbiter := 0; (* iteração atual *)
Melhiter := 0; (* melhor solução foi encontrada *)
Melhsol := s; (* melhor solução global*)
T := ∅;
Inicializar Função Critério de Aspiração A;
Enquanto (f(s) > f*) ou (Nbiter - Melhiter < Nbmax) faça
    Nbiter := Nbiter + 1
    Gerar um conjunto V* de soluções Si em N(s) o qual não é
    tabu ou que A(f(s)) >= f(si);
    Escolha uma solução s* minimizando f de V*;
    Atualize a Função critério de Aspiração A e a Lista Tabu T;
    Se (f(s*) < f (Melhsol)) então
        Melhsol := s*;
        Melhiter := Nbiter;
    Fim_se;
    s := s*;
Fim_enquanto;

```

onde :

X: conjunto de soluções possíveis;

f: Função objetivo;

N(s): a vizinhança da solução s em X;

T: Lista Tabu;

V\*: soluções vizinhas geradas em cada iteração;

f\*: o mais baixo limite da função objetivo f;

A: Função critério de Aspiração

A Busca Tabu realiza uma busca no espaço da mesma maneira que uma heurística de busca local comum, procedendo iterativamente de uma solução S para outra S' até que um critério de parada seja satisfeito. O que a diferencia de uma heurística de busca local comum é o emprego de uma estratégia de modificação da vizinhança N(S), proporcionando a troca de vizinhança N(S) por outra  $\overline{N(S)}$  conforme a busca progride. Como já foi dito, a Busca Tabu usa estruturas de memória de curto prazo e longo prazo para determinar  $\overline{N(S)}$  e organizar a busca. As soluções admitidas para  $\overline{N(S)}$  são determinadas pela identificação de soluções encontradas sobre um horizonte específico. Atributos que mudam durante este horizonte são armazenados em uma ou mais listas e recebem status de proibidos. As estruturas de memória de curto prazo gerenciam o que entra e sai das listas e explicitamente identifica o status destes atributos. Soluções que contêm o status proibido se tornam indisponíveis por um certo tempo. Um elemento importante da flexibilidade na Busca Tabu é a introdução de soluções proibidas para N(S) através do Critério de Aspiração. O critério de status proibido pode deixar de ser levado em conta se uma solução tabu tiver uma solução melhor do que qualquer outra já previamente vista

#### 4.1.3.1 Trabalhos de PRV com Busca Tabu

Essa área de pesquisa tem apresentado um grande crescimento desde a década de 90 produzindo algumas das heurísticas mais eficientes e flexíveis (Cordeau *et al*, 2002) para o PRV. A Busca Tabu é considerada como a melhor metaheurística para o PRV (Cordeau *et al*, 2002; Toth e Vigo, 2003; Prins, 2004; Xu e Kelly, 1996). Segundo Toth e Vigo (Toth e Vigo, 2003) as abordagens baseadas em Busca Tabu são capazes de determinar soluções de alta qualidade para o PRV. Por exemplo, ao considerar, o amplamente utilizado, conjunto de 14 problemas de Christofides e Eilon (1969), a maioria dos melhores resultados foi obtida através de soluções baseadas na Busca Tabu (Toth e Vigo, 2002).

O trabalho de Osman (1993) foi uma das primeiras implementações de sucesso da Busca Tabu para o PRV. Neste trabalho a vizinhança é definida através de um mecanismo de geração  $\lambda$ -*interchange*, com  $\lambda = 2$ . Os movimentos são gerados através de procedimento de 2-opt, realocação de vértices em diferentes rotas e troca de vértices. Osman propõe duas versões do algoritmo. Em uma versão, depois que toda a vizinhança foi explorada, o melhor movimento é aceito, na segunda versão, nem toda a vizinhança precisa ser explorada uma vez que é considerado como aceito o primeiro movimento encontrado.

No trabalho de Gendreau, Hertz e Laporte (1994), o algoritmo denominado Taburoute apresenta algumas características que o tornam mais robusto. Uma das características inovadoras é permissão do uso de soluções inviáveis com relação à capacidade do veículo ou duração da rota a fim de explorar outras regiões do espaço de busca. A maneira como os autores permitiram o uso destas soluções inviáveis é através da introdução de termos de penalidade na função objetivo. A vizinhança  $x_t$  é definida como sendo o conjunto de todas as soluções alcançáveis de  $x_t$  através da remoção do vértice  $v$  da sua rota atual e inserção dele em outra rota  $s$  contendo um dos seus vizinhos mais próximos. Como atributo tabu controla-se a quantidade de vezes que um vértice deve ficar sem ser utilizado. Esta quantidade é dada pelo valor  $\theta$  que é um valor randômico entre



[5,10]. Uma estratégia de diversificação que penaliza os vértices que são frequentemente movimentados é utilizada.

O algoritmo de Taillard (1993) apresenta algumas das características existentes no Taburoute, como a duração randômica do atributo tabu e utilização de técnicas de diversificação. A inovação apresentada por este trabalho se refere à decomposição do problema principal em subproblemas (Laporte *et al*, 2000; Cordeau *et al*, 2002) e cada subproblema é então resolvido independentemente em processadores paralelos. A estrutura de vizinhança é definida através de mecanismos de  $\lambda$ -*interchance*, com  $\lambda = 1$ . A re-otimização de rotas individuais é feita através de algoritmos exatos construídos para o PCV.

O trabalho de Xu e Kelly (1996) apresenta uma estrutura de vizinhança mais sofisticada. Ele considera trocas entre vértices de duas rotas, uma reposição global de alguns vértices em outras rotas e melhoramentos locais. A estratégia de reposição global resolve um modelo de fluxo de redes para realocar de maneira ótima um dado número de vértices em diferentes rotas. As políticas de intensificação e diversificação são baseadas em procedimentos de reinício de busca e em frequência de memória de longo termo respectivamente. Para o funcionamento da estratégia de intensificação, um *pool* de melhores soluções é mantido e usado para reiniciar a busca. Quando a busca é reiniciada, a solução atual é recuperada deste *pool*, as restrições tabus são relaxadas e os parâmetros reinicializados. A estratégia de diversificação é baseada em penalidades que são associadas aos vértices em função à frequência que eles são designados a rotas específicas. O objetivo é dar preferência aos vértices que aparecem com menor frequência nas rotas.

O algoritmo de Rego e Roucairol (1996) define a estrutura de vizinhança através da utilização de cadeias de ejeção para mover-se de uma solução a outra. Segundo os autores, uma interpretação genérica do termo consiste em sucessivamente selecionar um conjunto de elementos que são designados a um novo estado, de tal maneira que a cada passo a mudança do estado é criada pelo elemento do passo imediatamente precedente. Na terminologia de cadeias de ejeção,

cada passo é chamado nível e a mudança do estado de um elemento produzido por um outro é chamado uma ejeção. Desta maneira uma cadeia de ejeção pode ser vista como uma série de níveis  $l$ , cada um consistindo de três vértices  $(u_{s-1}, u_s, u_{s+1})$  que aparecem consecutivamente em uma rota. Um  $l$ -nível da cadeia de ejeção consiste em substituir a tripla  $(u_{s-1}^k, u_s^k, u_{s+1}^k)$ , onde  $k = 0, \dots, l$  pela tripla  $(u_{s-1}^{k+1}, u_s^{k+1}, u_{s+1}^{k+1})$ , onde  $k = 1, \dots, l$  até que o vértice posicionado no último nível,  $u_s$  seja realocado.

Rochat e Taillard (1995) apresentam o conceito de Memória Adaptativa. Uma memória adaptativa é um *pool* de boas soluções produzidas por heurísticas que é dinamicamente atualizado através da adição de novos elementos de alta qualidade e remoção de elementos de menor qualidade (Cordeau *et al*, 2002). Periodicamente, alguns elementos das soluções são extraídos e combinados diferentemente para produzir novas soluções (Laporte *et al*, 2000).

O trabalho de Barbarosoglu e Ozgur (1999) descreve um algoritmo de Busca Tabu usando apenas estratégias de intensificação examinando somente soluções viáveis. Soluções vizinhas são definidas através de um esquema de  $\lambda$ -interchange que favorece vértices relativamente distantes do centróide de suas rotas correntes e perto do centróide de uma nova rota. Também faz a reotimização de rotas através da execução de procedimentos 2-opt (Gendreau *et al*, 1998).

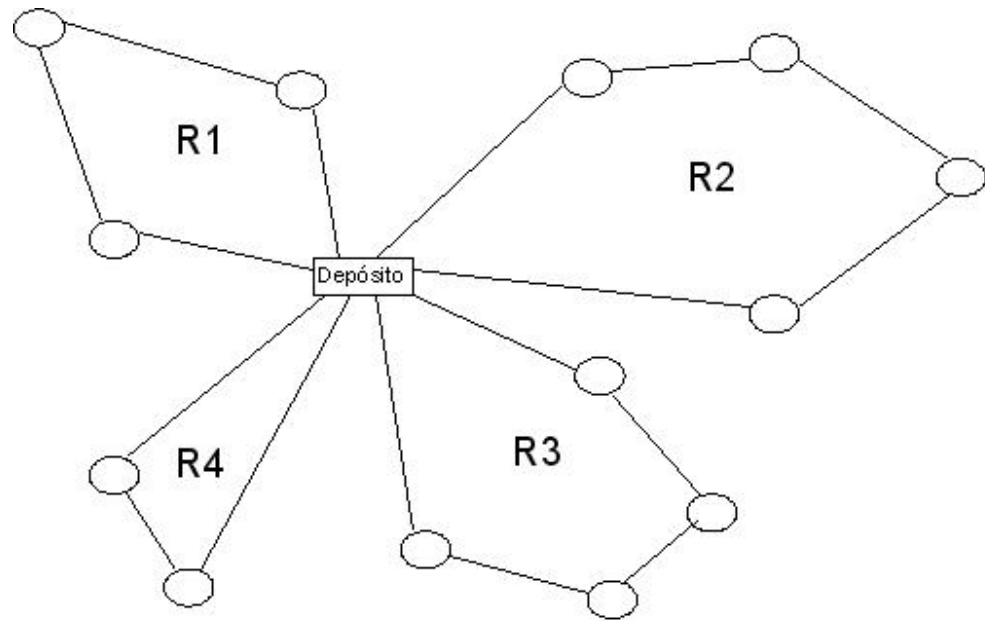
Toth e Vigo (2003) apresentam um algoritmo chamado *Granular Tabu Search* em que a idéia principal é desconsiderar as áreas não promissoras do espaço de busca através da remoção de arcos de alto custo quando o problema é modelado em um grafo completo. Um limite associado ao tamanho máximo dos arcos esperados em uma solução de alto nível é definido e desta maneira arcos superiores a este limite são eliminados. O intuito é restringir drasticamente o espaço de soluções possíveis removendo elementos que tem baixa probabilidade de pertencerem a soluções de boa qualidade. Soluções inviáveis também são permitidas através do uso de penalidades associadas ao excesso de demanda e distância da solução.

## 5 MODELO PROPOSTO

Este capítulo tem por objetivo apresentar o modelo definido neste trabalho. A Seção 6.1 apresenta o problema utilizado. A seção 6.2 apresenta formulação utilizada e a seção 6.3 apresenta a arquitetura da aplicação desenvolvida.

### 5.1 DEFINIÇÃO DO MODELO

O modelo gerado neste trabalho foi desenvolvido utilizando a versão clássica do problema, o PRVC, em que a única restrição existente se refere à capacidade do veículo. Nesta versão do problema assume-se que todos os veículos têm capacidade homogênea e estão localizados em um único depósito. Existe uma demanda conhecida para cada cliente. Todas as rotas devem iniciar e terminar no depósito. A demanda total de uma rota não pode exceder a capacidade do veículo. O objetivo é criar um conjunto de rotas que atenda todos os clientes minimizando os custos deste atendimento. Uma vez que este problema é *NP-Hard* optou-se por desenvolver o modelo utilizando a metaheurística Busca Tabu. A Busca Tabu, apresentada por Glover (1986), é uma técnica que oferece recursos capazes de tornar a procura de uma boa solução para o PRV robusta e flexível. Com o uso da Busca Tabu procura-se ter flexibilidade nos processos de busca no espaço, através das definições dos movimentos e na geração de vizinhanças, tal que, pela diversidade de soluções geradas obtenha-se uma boa qualidade do resultado final obtido. E segundo diversos autores (Breedam, 2001; Baker e Ayechev, 2003; Cordeau *et al*, 2002; Toth e Vigo, 2003; Prins, 2004; Xu e Kelly, 1996) a Busca Tabu é uma das metaheurísticas de maior sucesso na resolução do PRV. A estrutura de uma solução para o problema abordado neste modelo é ilustrada na figura 14.



**FIGURA 14** – Ilustração de uma possível solução do PRV: um conjunto de rotas que atende todos os clientes.

## 5.2 FORMULAÇÃO MATEMÁTICA

A formulação utilizada neste modelo é uma simplificação da apresentada por Barbarasoglu e Ozgur (1999), uma vez que a restrição de tempo de duração da rota não é aqui utilizada. Assim sendo, a formulação utilizada já com a modificação é apresentada a seguir:

$$\text{Minimizar } \sum_i \sum_j \sum_v c_{ij} X_{ij}^v; \quad (6.1)$$

sujeito a:

$$\sum_i \sum_v X_{ij}^v = 1, \text{ para todo } j; \quad (6.2)$$

$$\sum_j \sum_v X_{ij}^v = 1, \text{ para todo } i; \quad (6.3)$$

$$\sum_i X_{ip}^v - \sum_j X_{pj}^v = 0, \text{ para todo } p, v; \quad (6.4)$$

$$\sum_i d_i \left( \sum_j X_{ij}^v \right) \leq Q_v, \text{ para todo } v; \quad (6.5)$$

$$\sum_{j=1}^n X_{0j}^v \leq 1, \text{ para todo } v; \quad (6.6)$$

$$\sum_{i=1}^n X_{i0}^v \leq 1, \text{ para todo } v; \quad (6.7)$$

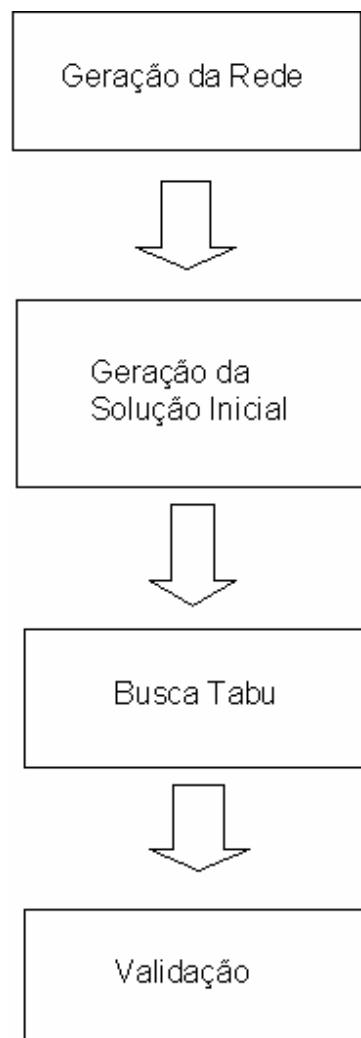
$$X_{ij}^v \in Z, \text{ para todo } i, j \text{ e } v; \quad (6.8)$$

onde  $X_{ij}^v$  são variáveis binárias que indicam se o arco  $(v_i, v_j)$  é utilizado pelo veículo  $v$ . A função objetivo de minimização da distância/custo/tempo aparece na equação (6.1). As restrições representadas nas equações (6.2) e (6.3) juntas garantem que cada vértice é atendido por um único veículo. A equação (6.4) garante que um veículo deixa o vértice aonde foi atender a demanda tão logo alcance este vértice. A capacidade do veículo  $Q_v$  é expressa na equação (6.5) aonde  $d_i$  representa a demanda de cada vértice  $i$ . As restrições (6.6) e (6.7) expressam que a disponibilidade do veículo não pode ser excedida. A equação (6.8) representa a eliminação de *subtour* onde  $Z$  pode ser definido por:

$$Z = \left\{ (X_{ij}^v): \sum_{i \in B} \sum_{j \in B} X_{ij}^v \leq |B| - 1 \text{ para } B \subseteq V / \{0\}; |B| \geq 2 \right\}.$$

### 5.3 ARQUITETURA DO MODELO

Para implementar este modelo foi definida uma arquitetura composta de 4 módulos: módulo de geração das redes, módulo de geração da solução inicial, módulo da Busca Tabu e módulo de validação. A figura 15 ilustra a arquitetura proposta.



**FIGURA 15** - Arquitetura do Modelo.

O módulo de Geração da Rede, através dos dados (vértices, coordenadas e demandas) obtidos a partir de instâncias clássicas selecionadas, gera uma matriz de distância entre todos os vértices da rede. A matriz de distância é utilizada no módulo de geração da Solução Inicial para gerar a solução que será utilizada no módulo da Busca Tabu. A solução inicial é gerada a partir da Heurística do Vizinheiro mais Próximo (Tyagi, 1968). No módulo da Busca Tabu, a partir de uma solução inicial viável, o algoritmo da Busca Tabu percorre o espaço de busca, através da geração de soluções vizinhas e escolha da melhor solução de cada vizinhança. Além disso, durante o processo de busca é utilizada uma estratégia de intensificação buscando melhorar a qualidade das soluções geradas. A seguir, o detalhamento dos módulos que compõe arquitetura.

### 5.3.1 Módulo de Geração da Rede

Este módulo gera a matriz de distância que será utilizada no modelo. Instâncias clássicas de PRV foram selecionadas da literatura e as informações referentes aos seus vértices, coordenadas e demandas são utilizadas para a criação das matrizes.

As matrizes de distância são criadas da seguinte maneira: Para cada par de vértices com coordenadas  $(x_1, y_1)$ ,  $(x_2, y_2)$  é calculada a distância euclidiana entre eles através da fórmula  $\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$  (Cook et al, 1998). O resultado deste módulo é uma matriz simétrica com as distâncias euclidianas entre quaisquer dois vértices para cada instância.

A figura 16 representa uma matriz de distância gerada neste módulo

Matriz de Distância da instância X									
	1	2	3	4	5	6	...		
1		$d_{12}$	$d_{13}$	$d_{14}$	$d_{15}$	$d_{16}$			
2	$d_{21}$		$d_{23}$	$d_{24}$	$d_{25}$	$d_{26}$			
3	$d_{31}$	$d_{32}$		$d_{34}$	$d_{35}$	$d_{36}$			
...									

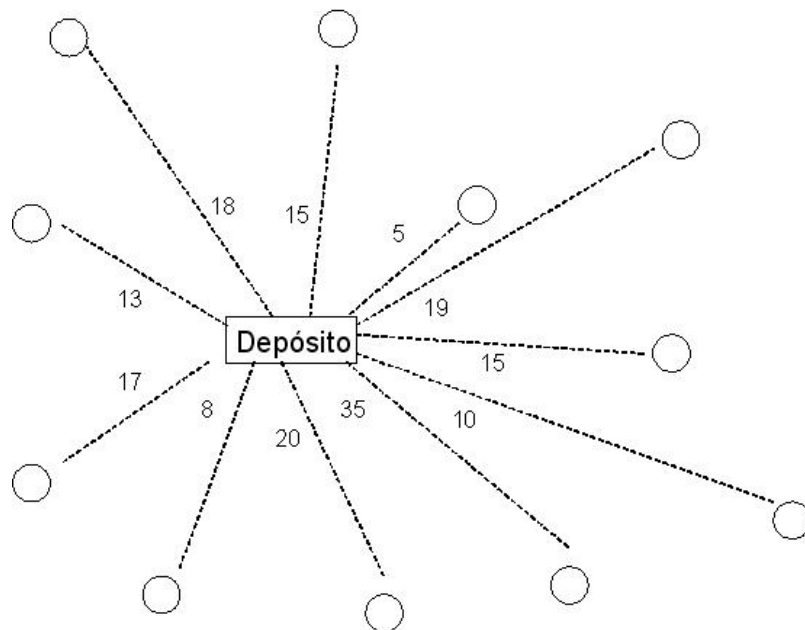
FIGURA 16 – Matriz de Distância.

### 5.3.2 Módulo da Geração da Solução Inicial

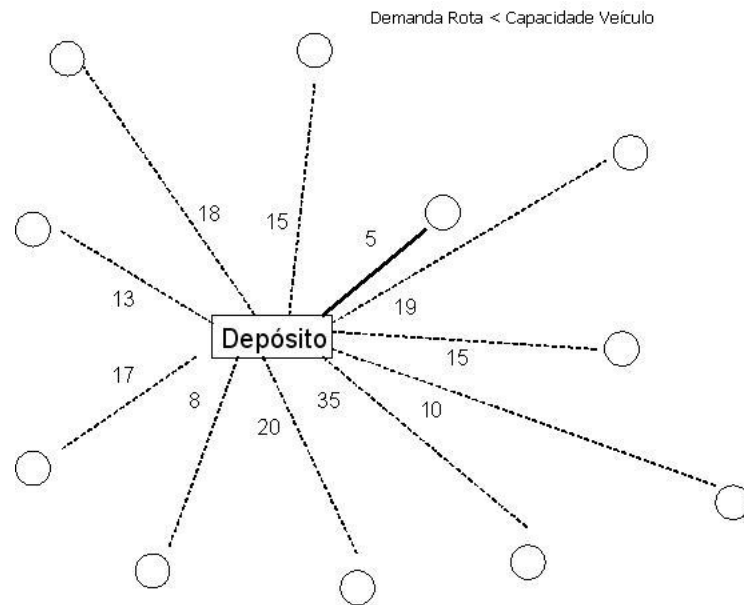
Este módulo utiliza a matriz de distância obtida no módulo anterior para gerar uma Solução Inicial para o problema. Será utilizado o Método do Vizinho mais Próximo, apresentado na seção 5.1.2.5.1 para gerar a solução inicial. Nesta heurística o próximo vértice a ser inserido na rota é determinado pelo último que foi inserido, uma vez que segundo esta heurística, o próximo vértice selecionado é sempre o mais próximo do último vértice inserido. Como as rotas devem iniciar e terminar no depósito, naturalmente, o primeiro arco escolhido será um arco  $(v_0, v_i)$  em que  $v_0$  representa o depósito e  $v_i$  um outro vértice qualquer em que o valor deste arco seja mínimo. Dado que a capacidade do veículo é o limitante da quantidade de clientes que podem ser inseridos numa rota, em função das demandas associadas, toda vez que a inserção de um próximo cliente à rota violar a restrição da capacidade, este cliente não é inserido e retorna-se ao depósito. Se ainda houver clientes não atendidos, a partir do depósito inicia-se a formação de uma nova rota, desprezando-se os vértices e arcos já utilizados. A solução inicial gerada neste módulo é representada por uma rede que interliga todos os vértices representando os clientes através de rotas que iniciam e terminam no vértice que representa o depósito. A figura 17a apresenta as distâncias entre os vértices e o depósito. Na figura 17b, o vértice mais próximo do



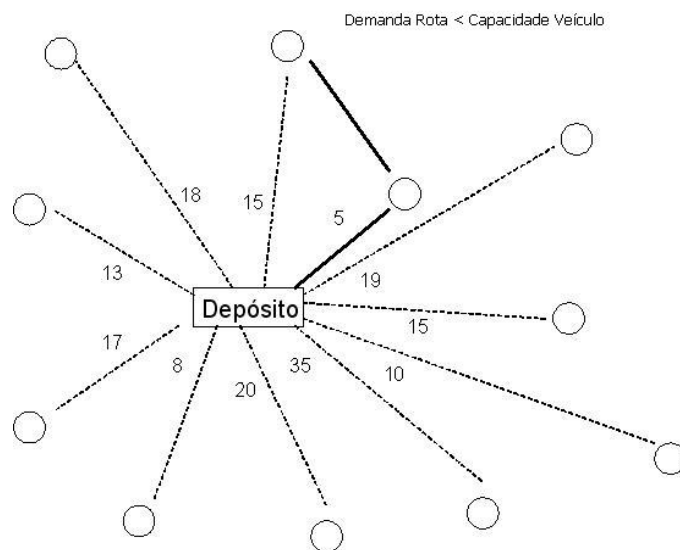
depósito foi selecionado. As figuras 17c e 17d ilustram a formação da rota com a adição dos vértices mais próximos ao último que foi inserido na rota. Enquanto a demanda da rota for menor que a capacidade do veículo, este procedimento é realizado. A figura 19e representa o caso em que a adição do vértice mais próximo ao último provocaria um excesso de demanda na capacidade do veículo, portanto, a rota deve ser concluída, com o veículo retornando ao depósito e um novo vértice deve ser selecionado para iniciar a formação de outra rota, no caso, o vértice mais próximo do depósito foi o escolhido.



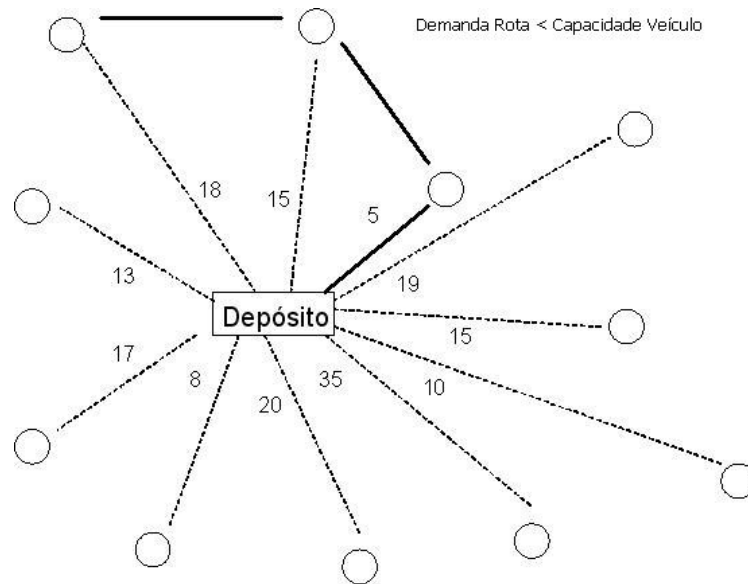
**FIGURA 17a** – Rede com a Distância entre o depósito e todos os clientes.



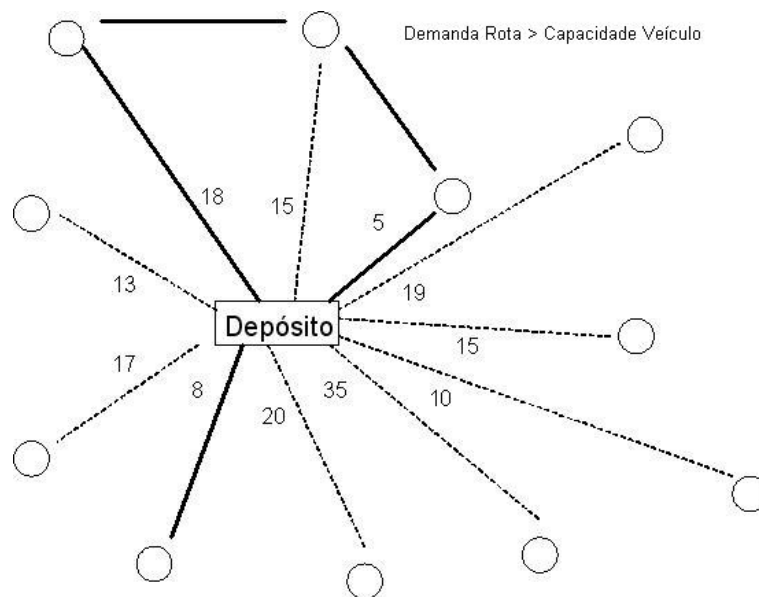
**FIGURA 17b** – Início da formação de uma rota a partir da seleção do cliente mais próximo ao depósito.



**FIGURA 17c** – Adição de um segundo cliente à rota.



**FIGURA 17d** – Adição de um terceiro cliente à rota.



**FIGURA 17e** – Retorno ao depósito do veículo que atende a rota e início da formação de uma nova rota.

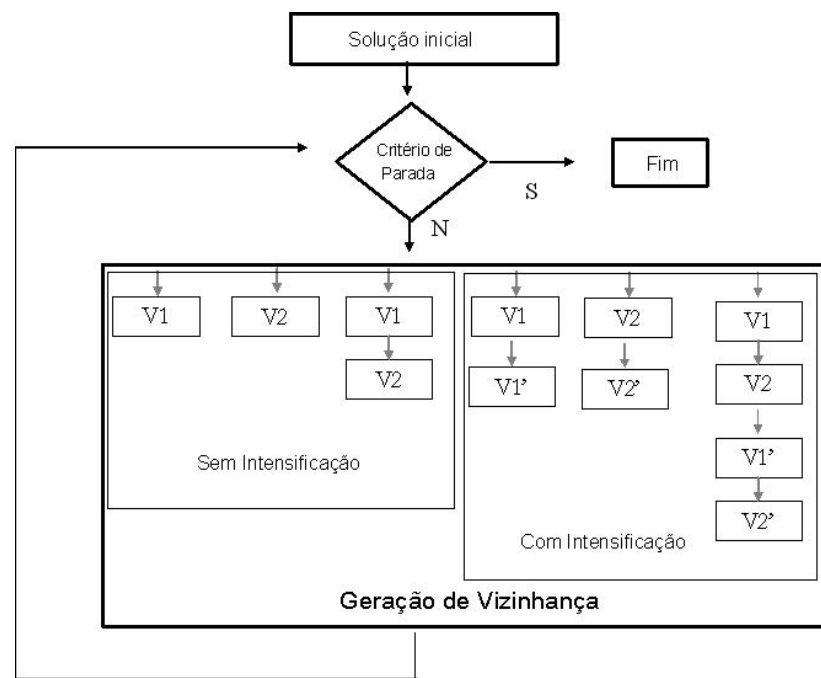
### **5.3.3 Módulo da Busca Tabu**

Este módulo implementa o algoritmo da Busca Tabu. Neste item serão abordados a dinâmica do algoritmo, os movimentos geradores de vizinhança, a estratégia de intensificação, a Lista Tabu, o Critério de Aspiração e o Critério de Parada.

#### **5.3.3.1 Dinâmica do Algoritmo da Busca Tabu**

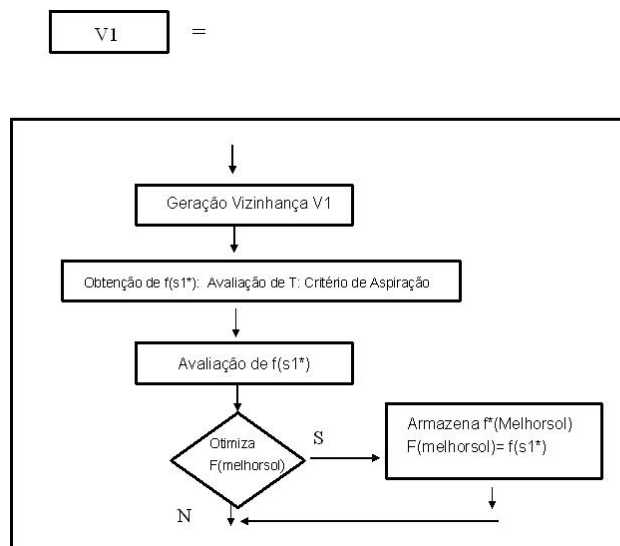
A partir de uma solução inicial, movimentos são executados gerando um conjunto de novas soluções, denominado de vizinhança da solução. Desta vizinhança, a Busca Tabu escolhe a melhor solução e a partir dela gera um novo conjunto de soluções. Neste modelo, duas políticas dão origem aos movimentos: realocação de vértices e a troca de vértices entre rotas. Baseado nestas duas políticas 4 movimentos foram definidos: V1, V2, V1' e V2'. Os movimentos V1 e V2 são utilizados durante o processo normal da busca. Os movimentos V1' e V2' são utilizados em conjunto com V1 e V2, respectivamente, no processo de busca com intensificação. A dinâmica destes movimentos é detalhada no item 6.3.3.5. A Busca Tabu executa o que Glover e Laguna (1997) chamam de exploração agressiva do espaço de busca. Após a geração de uma vizinhança, a Busca Tabu escolhe como próxima solução àquela que apresentar o melhor valor para a função objetivo, independente de esta solução melhorar ou não o valor atual da função objetivo. Para ajudar na escolha da melhor solução uma estrutura auxiliar, chamada Lista Tabu é utilizada. A Lista Tabu guarda atributos que são considerados proibidos e que não devem aparecer na solução atual. Desta forma, a Lista Tabu é utilizada para impedir que soluções com atributos proibidos sejam escolhidos como próxima solução. Essa propriedade da Lista Tabu também é utilizada para evitar que a busca fique presa em um mínimo local do espaço de busca. O tamanho da Lista Tabu define a quantidade de iterações que aqueles atributos não devem ser utilizados. Caso a solução escolhida seja Tabu, ela ainda poderá ser utilizada se o Critério de Aspiração puder ser aplicado. O Critério de Aspiração permite que uma solução proibida seja perdoada e possa ser utilizada caso o valor da sua função objetivo seja melhor do que o resultado da melhor solução

vigente. Este procedimento segue até que o Critério de Parada seja encontrado. A dinâmica do algoritmo da Pesquisa Tabu implementado neste modelo é representado na figuras 18 e 19.



**FIGURA 18** - Estrutura geral do algoritmo da Pesquisa Tabu implementado no modelo proposto neste trabalho.

A figura 19 a seguir ilustra a expansão de um movimento gerador de vizinhança, detalhando o algoritmo da Busca Tabu.



**FIGURA 19** – Detalhamento do algoritmo da Pesquisa Tabu com a expansão de V1.

### 5.3.3.2 Solução Inicial

A Solução Inicial é gerada através da heurística do Vizinho mais Próximo, conforme explicado na seção 6.3.2. Uma vez que esta solução está disponível inicia-se o processo de busca de uma solução que otimize a função objetivo definida no modelo (item 6.2).

### 5.3.3.3 Lista de Soluções Elite

A Lista de Soluções Elite é uma lista que armazena apenas as melhores soluções encontradas ao longo do processo de busca. Neste modelo, a Lista de Soluções Elite é criada com a finalidade de permitir a aplicação da estratégia de intensificação. A lista é atualizada a cada nova melhor solução encontrada. Quando a lista estiver cheia, cada novo resultado substituirá o resultado com maior valor da função objetivo. Foi definido um tamanho máximo de 100 soluções para serem armazenadas na lista.

#### **5.3.3.4 Estratégia de Intensificação**

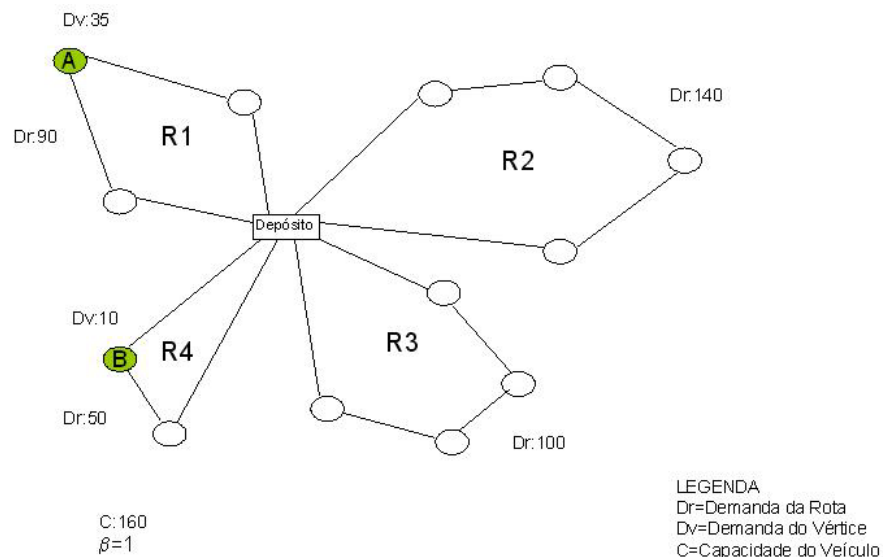
Uma estratégia de intensificação tem por objetivo focar a busca em soluções historicamente consideradas boas. A estratégia utilizada nesta aplicação visa retornar as melhores soluções armazenadas na Lista de Soluções Elite e gerar novas vizinhanças expandidas daquelas soluções. A cada 15 iterações sem melhora na solução global, aplica-se a estratégia de intensificação. O objetivo é pesquisar se naquela vizinhança expandida existe uma solução melhor do que a solução escolhida quando a busca passou por aquela região anteriormente. A melhor solução encontrada a partir da nova vizinhança expandida gerada é escolhida para a continuação do processo da busca.

#### **5.3.3.5 Movimentos Geradores de Vizinhança**

Neste modelo serão utilizados movimentos geradores de vizinhança baseados em duas políticas: a realocação de vértices e a troca de vértices. Basicamente, o movimento de troca de vértices consiste em selecionar dois vértices de rotas diferentes e trocá-los de rotas e o movimento de realocação de vértices consiste em retirar um vértice de uma rota origem e o inserir em uma rota destino. Espera-se com o uso destas políticas proporcionar uma diversificação das soluções geradas pelos movimentos. Baseados nestas duas políticas foram definidos os seguintes movimentos:

(V1): Implementa a política de troca de vértices. Uma rota é selecionada para ser a origem dos vértices. Para cada vértice desta rota testa-se se é possível sua troca com todos os outros vértices de todas as outras rotas. Uma troca só é possível se não gerar excesso na capacidade dos veículos que atendem as duas rotas com a adição das novas demandas nas rotas. Cada troca realizada gera um novo vizinho. As figuras 20a, 20b, 20c e 20d ilustram a dinâmica deste movimento. A rota R1 é selecionada para ser a origem dos vértices. O procedimento é ilustrado apenas para o vértice A, entretanto o movimento é realizado para todos os vértices da R1. A figura 20a mostra a seleção do vértice A da R1 e do vértice B da R4. A

figura 20b ilustra a troca realizada, o vértice A passa a ocupar o lugar do vértice B em R4 e vice-versa. A figura 20c, mostra a seleção do vértice A da R1 e do vértice C da R4. A troca de posição dos vértices é ilustrada na figura 20d. Na continuação deste movimento essa dinâmica da troca seria realizada também para todos os outros vértices da rota R1.



**FIGURA 20a** – Dinâmica de V1: Seleção do vértice A na rota origem R1 e do primeiro vértice a ser trocado na rota destino R4.



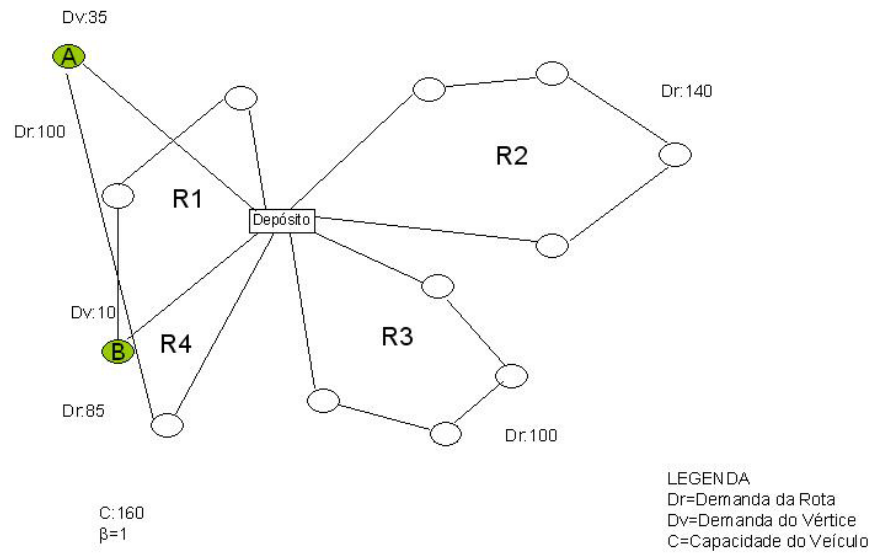


FIGURA 20b - Dinâmica de V1: Troca entre os vértice A e B.

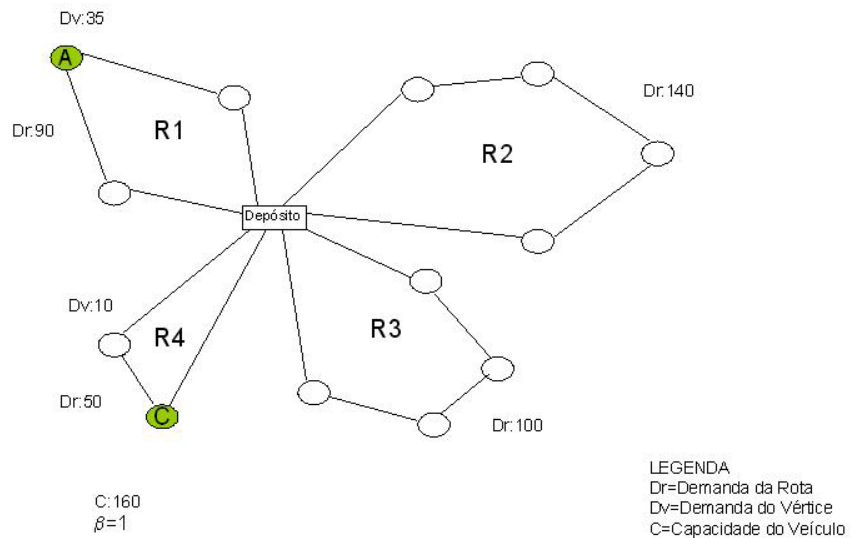
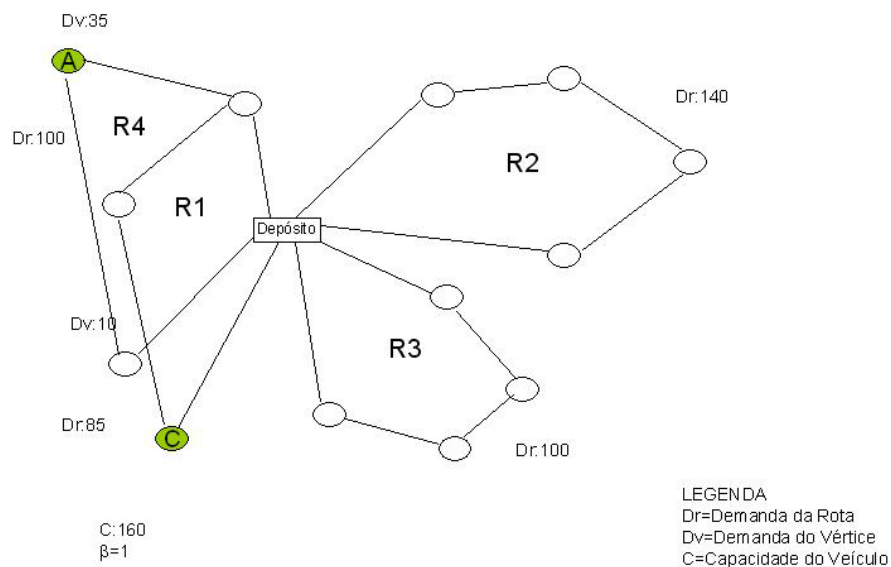


FIGURA 20c - Dinâmica de V1: Seleção do vértice A na rota origem R1 e do segundo vértice a ser trocado na rota destino R4.



**FIGURA 20d** - Dinâmica de V1: Troca entre os vértice A e C.

V2: Implementa a política de realocação de vértices. Uma rota é escolhida como rota origem e então todos os vértices desta rota, um por vez, serão realocados nas outras rotas existentes, chamadas de rotas destino, caso a restrição da capacidade do veículo seja respeitada. Para cada vértice da rota origem, depois de encontrada uma rota destino, insere-se o vértice em todas as possíveis posições da rota destino, gerando para cada posição um uma nova solução vizinha. As figuras a seguir ilustram a dinâmica deste movimento. Na figura 21a o vértice A com demanda = 35 foi selecionado na rota R1. Nas figuras 21b, 21c e 21d o vértice A foi inserido na rota R4 em todas as possíveis posições. A inserção nesta rota é permitida uma vez que a demanda do vértice A somada a demanda da rota R4 não ultrapassa a capacidade do veículo. Na continuação deste movimento a inserção seria realizada também para a rota R3. A única rota em que o vértice A não pode ser inserido é na rota R2, pois sua inserção resultaria num estouro da capacidade do veículo ( $140 + 35 = 175$ )

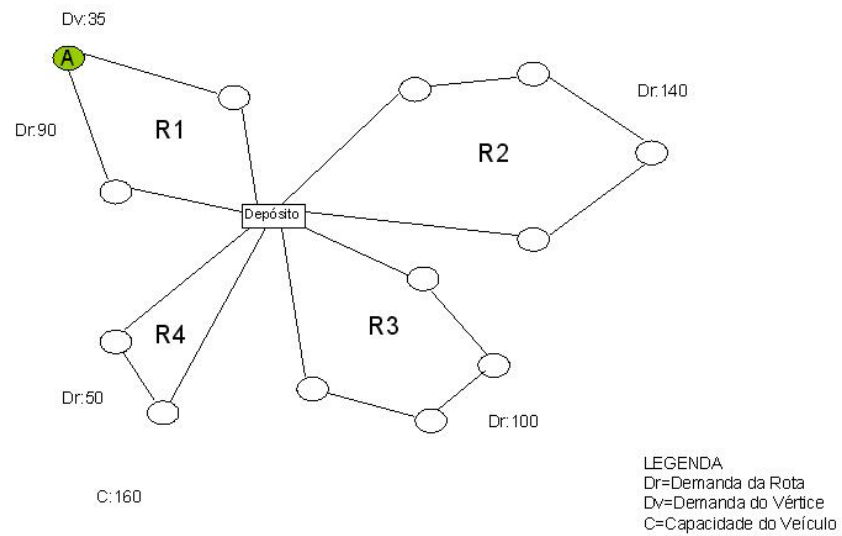


FIGURA 21a – Dinâmica de V2: seleção da rota origem R1 e do vértice A.

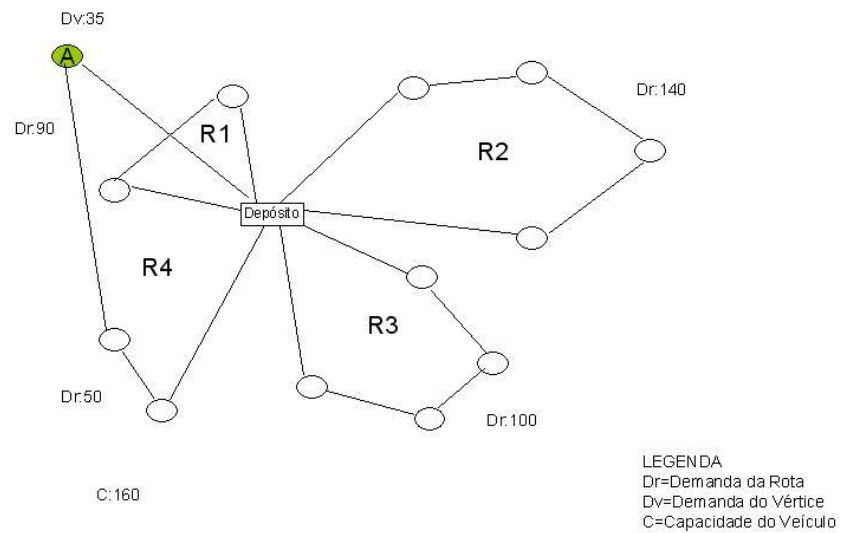
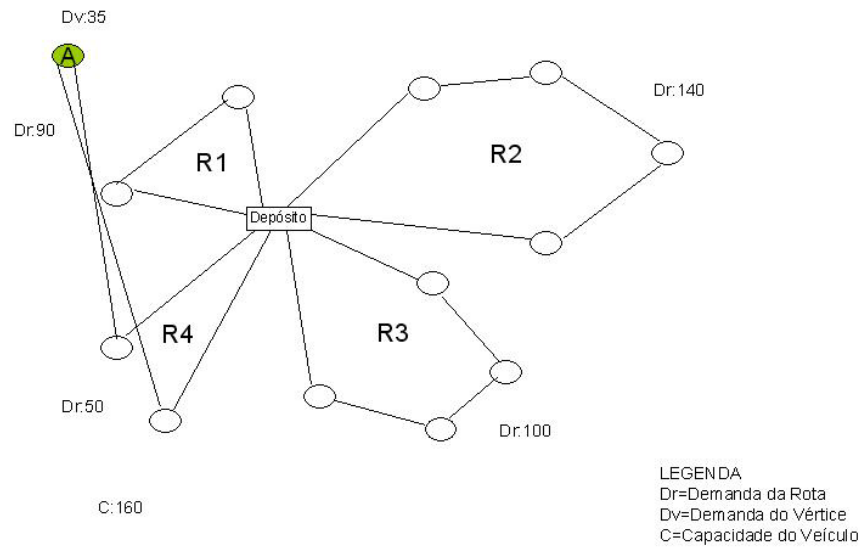
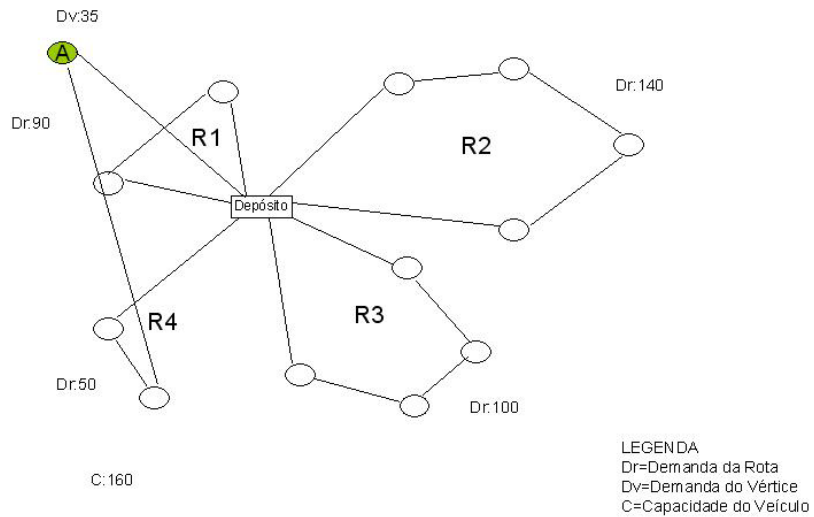


FIGURA 21b - Dinâmica de V1: seleção de uma rota destino, R4, e inserção de A na primeira posição possível.



**FIGURA 21c** - Dinâmica de V2: inserção de A na segunda posição possível da rota R4.



**FIGURA 21d** - Dinâmica de V2: inserção de A na terceira posição possível da rota R4.

V1': Este movimento apresenta a mesma dinâmica do V1. A diferença entre os movimentos V1 e V1' consiste na quantidade de rotas que são escolhidas como origem, para terem seus vértices trocados. Enquanto no V1, apenas uma rota é selecionada, no V1', todas as rotas são selecionadas para terem seus vértices trocados. A denominação V1' foi criada para facilitar a leitura do trabalho. Ao longo da leitura do mesmo é possível encontrar referência a este movimento (V1') quando no texto são feitas considerações sobre o processo de busca utilizando *V1 com intensificação*.

V2': Apresenta a mesma dinâmica do movimento V2. A diferença entre os movimentos V2 e V2' consiste na quantidade de rotas que são escolhidas como origem, para terem seus vértices realocados. Enquanto no V2, apenas uma rota é selecionada, no V2', todas as rotas são selecionadas para terem seus vértices trocados. Semelhante ao movimento anterior, referência a este movimento (V2') é feita quando no texto aparecem considerações sobre o processo de busca utilizando *V2 com intensificação*.

### **5.3.3.6 Lista Tabu**

A Lista Tabu é uma estrutura definida para guardar os movimentos considerados proibidos. Neste modelo, todo o conjunto de rotas que representa a solução escolhida é guardado na Lista Tabu. Um tamanho  $T$  deve ser definido para a Lista para representar a quantidade de iterações que os atributos devem ser proibidos. Assim, aquela solução que está na Lista Tabu, fica proibida de ser usada por  $T$  iterações. Nesta aplicação, o tamanho  $T$  é um valor fixo para cada iteração. Caso a solução escolhida seja proibida, esta solução poderá ser aceita se for admitida pelo Critério de Aspiração.

#### **5.3.3.7 Critério de Aspiração**

O Critério de Aspiração utilizado é o critério clássico que corresponde a permitir que uma solução tabu seja aceita apenas se ele tiver o valor da função objetivo melhor do que o valor da melhor solução vigente.

#### **5.3.3.8 Critério de Parada**

O Critério de Parada adotado corresponde ao critério clássico, em que se define um número máximo de iterações ( $Nb_{max}$ ) que podem ocorrer sem que haja uma melhora no valor da função objetivo na solução atual.

#### **5.3.4 Módulo da Validação**

Neste módulo é feita a validação dos resultados gerados através da comparação com outros resultados já publicados. Foram selecionados heurísticas clássicas e trabalhos que utilizam a Busca Tabu para serem utilizadas na comparação.

## 6 EXPERIMENTOS E VALIDAÇÃO DO MODELO

Neste capítulo é apresentado o planejamento dos experimentos realizados, bem como os resultados obtidos. A seção final do capítulo apresenta a validação do modelo através da comparação dos resultados obtidos com resultados de outros trabalhos publicados.

### 6.1 PLANEJAMENTO DOS EXPERIMENTOS

O objetivo de se realizar experimentos com a Busca Tabu é encontrar soluções de boa qualidade através da calibragem dos valores para o tamanho de Lista Tabu, número de iterações sem melhora na função objetivo ( $Nb_{max}$ ) e tipos de movimentos geradores de vizinhança. Sabendo que esses valores podem influenciar na qualidade das soluções, a experimentação de diferentes valores é o que permite ao pesquisador o desenvolvimento de uma aprendizagem sobre o comportamento do modelo. Os experimentos serão realizados para as instâncias: Problemas 1, 2, 3, 4 e 5 de Christofides, Mingozzi e Toth (1979), com 50, 75, 100, 150 e 199 clientes e 1 depósito respectivamente. Para a realização dos experimentos deste trabalho, as seguintes etapas serão consideradas:

-Obtenção de uma solução inicial para cada instância: para a execução do algoritmo da Busca Tabu é necessário informar uma solução inicial pra ser utilizada como entrada para a execução do algoritmo. As soluções iniciais para cada problema foram geradas seguindo a heurística do vizinho mais próximo conforme explicado no item 6.3.3.2 do capítulo anterior. Os valores das soluções iniciais obtidos foram: i)problema 1: 715,04 ii)problema 2:1077,75 iii)problema 3: 1054,71 iv)problema 4: 1380,80 v)problema 5:1643,45.

-Variação dos movimentos geradores de vizinhança: com o objetivo de tornar a busca mais flexível foram definidos dois movimentos geradores de vizinhança, a realocação de vértices e a troca de vértices entre rotas. O detalhamento desses movimentos é apresentado no item 6.3.3.5. do capítulo anterior.

-Combinação dos mecanismos geradores de vizinhança: com o objetivo de testar os diferentes movimentos, os movimentos serão utilizados isoladamente e depois associados.

-Variação do tamanho da Lista Tabu: foram definidos diversos valores para o tamanho da Lista Tabu com o objetivo de testar se o tamanho da Lista Tabu influencia na qualidade do processo de busca.

-Variação do *Nbmax*: semelhante ao tamanho da Lista Tabu, definiu-se diversos valores para o *Nbmax* com o objetivo de testar se a variação do tamanho do *Nbmax* influencia na qualidade do processo de busca.

-Utilização da estratégia de intensificação: é proposta uma estratégia de intensificação com o objetivo de verificar se o uso da estratégia de intensificação gera um impacto positivo na qualidade do processo de busca. Também será possível avaliar a qualidade dos movimentos geradores de vizinhança em função da melhora de qualidade obtida com o uso da intensificação.

Os valores propostos para o tamanho do *Nbmax*, Lista Tabu e os movimentos utilizados são apresentados a seguir. A tabela 2 apresenta os valores propostos para o tamanho da Lista Tabu, a tabela 3 apresenta uma relação para os valores do número máximo de iterações sem melhora na função objetivo, a tabela 4 apresenta os movimentos geradores de vizinhança.

**TABELA 2** - Relação dos valores para o Tamanho da Lista Tabu utilizados.

Tamanho da Lista Tabu
10
25
50
75
100
200



**TABELA 3-** Relação dos valores para o *Nbmax* utilizado.

Nbmax
100
250
500
750
1000
1250
1500
1750
2000

**TABELA 4 –** Combinação de movimentos utilizados.

Nome Procedimento	Movimento Utilizado
V1 sem intensificação	V1
V2 sem intensificação	V2
V1,V2 sem intensificação	V1,V2
V1 com intensificação	V1 e V1'
V2 com intensificação	V2 e V2'
V1,V2 com intensificação	V1,V2 e V1',V2'

A figura 24 ilustra a estrutura utilizada para armazenar os resultados dos experimentos gerados chamada de Tabela de Resultados. Conforme visto nas tabelas 2 e 3 foram definidos duas séries de valores, uma para o tamanho da Lista Tabu e outra para o tamanho do *Nbmax*. Para cada tamanho da Lista Tabu, uma execução do algoritmo será realizada com cada um dos valores de *Nbmax*. Esse procedimento será realizado em duas etapas. Uma etapa sem uso da estratégia de intensificação e outra etapa com uso da estratégia de intensificação. Com o intuito de aumentar o tamanho da vizinhança gerada e proporcionar maior diversificação das soluções, optou-se por utilizar os movimentos em conjunto. Na primeira etapa, o algoritmo utilizará os movimentos, V1

e V2 separados e V1 e V2 juntos. Na segunda etapa serão utilizados os movimentos V1 e V1', V2 e V2' separados e agrupados. Desta maneira, classificando pelos movimentos utilizados serão gerados 6 conjuntos de experimentos para cada problema: i) V1, ii)V2, iii)V1 e V2, iv)V1 e V1', v)V2 e V2', vi)V1 e V2 com V1' e V2'. Cada conjunto de experimentos será composto de 54 execuções do algoritmo. Para cada execução, um valor de Nbmax e um tamanho da Lista Tabu diferentes serão utilizados, gerando uma possível solução ou resultado do problema para aqueles valores de Nbmax e Lista Tabu.

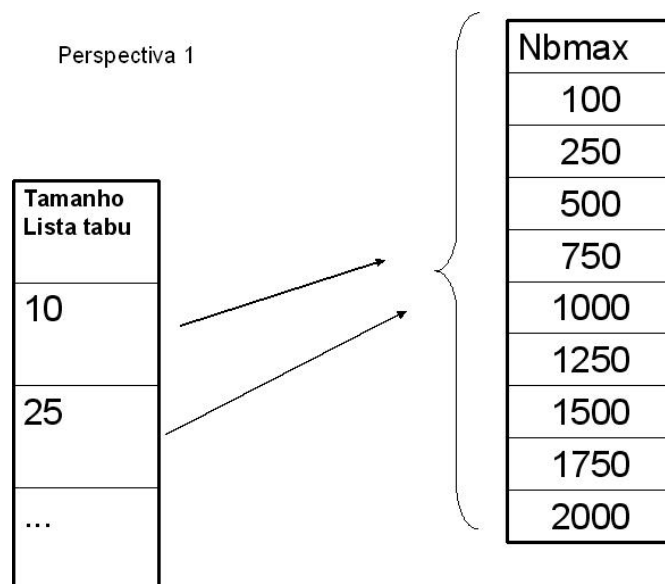
Problema:									
Lista	Nbmax								
Tabu	100	250	500	750	1000	1250	1500	1750	2000
10	Resul 1.1	Resul 1.2	(...)						
25	Resul 2.1								
50	(...)								
75									
100									
200									

**FIGURA 24** - Estrutura de armazenamento dos resultados dos experimentos.

Como cada conjunto contém 54 experimentos e serão realizados 6 conjuntos de experimentos para cada problema, no total serão 324 experimentos realizados. Para organizar a apresentação dos resultados no trabalho, os experimentos foram classificados sob duas perspectivas: i) Perspectiva 1 (Visão horizontal da Tabela de Resultados): variação do Nbmax para Lista Tabu, ou seja, para cada tamanho de Lista Tabu, testa-se todos os valores do Nbmax e ii) Perspectiva 2 (Visão vertical da Tabela de Resultados): variação do tamanho da Lista Tabu para cada valor do Nbmax, ou seja, a cada valor de Nbmax, testa-se todos os tamanhos da Lista Tabu. Com o intuito de gerar uma análise comparando os diversos resultados obtidos para os diferentes valores de Nbmax e tamanho da Lista Tabu, define-se como “melhor resultado”, aquele que apresenta o menor custo entre os resultados gerados para um dado valor fixo de Nbmax ou Lista Tabu. Analisando sob a Perspectiva 1, o melhor resultado é aquele que apresentar o menor custo para o tamanho da Lista Tabu e variação do Nbmax proposta. Na

Perspectiva 2, o melhor resultado é aquele que apresentar menor custo para aquele valor de Nbmax e variação do tamanho da Lista Tabu. O termo “melhor resultado” também é utilizado para qualificar o melhor resultado obtido para cada um dos 6 conjuntos de experimentos, neste caso, não considerando nenhum valor fixo de Lista Tabu ou Nbmax.

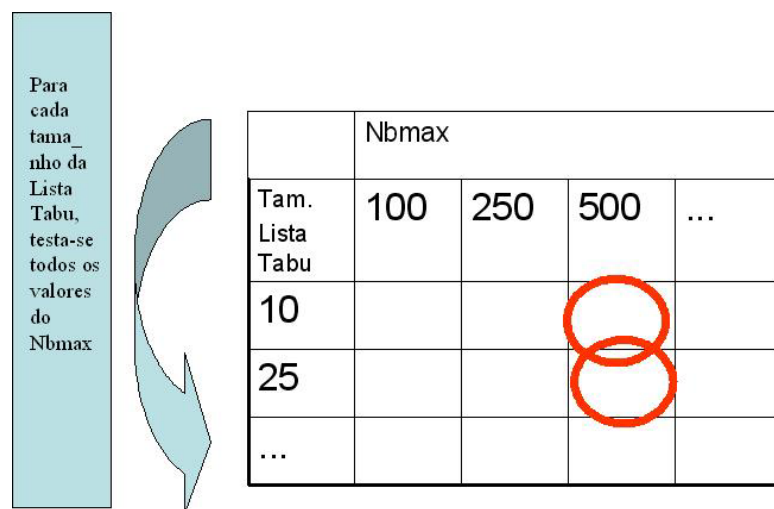
A figura 22a ilustra a relação entre o tamanho da Lista Tabu e Nbmax existente na perspectiva 1:



**FIGURA 22a** - Variação do valor do Nbmax para cada tamanho da Lista Tabu.

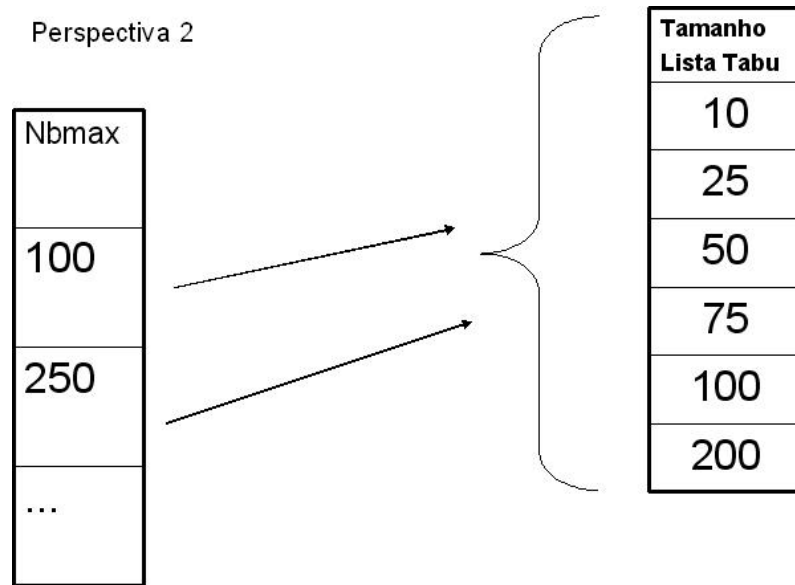
Na figura 22a é possível observar que para cada tamanho da Lista Tabu utilizam-se todos os valores do Nbmax. Para cada tamanho, o melhor resultado é aquele que apresentar o menor custo entre todos os resultados gerados com os diversos valores de Nbmax. A figura 22b a seguir, ilustra a localização de possíveis melhores resultados do conjunto. Conforme pode ser visto, sob a perspectiva 1, o melhor resultado é aquele que apresentar o menor custo entre todos os resultados gerados para aquele tamanho fixo de Lista Tabu e variação proposta do Nbmax. Desta maneira, na perspectiva 1, é possível haver mais de um melhor resultado para o mesmo valor de

Nbmax, entretanto para cada tamanho da Lista Tabu, apenas o resultado de menor custo é considerado o melhor. Sendo assim, de acordo com os valores propostos para o tamanho da Lista Tabu é possível gerar 6 melhores resultados para cada conjunto de movimentos.



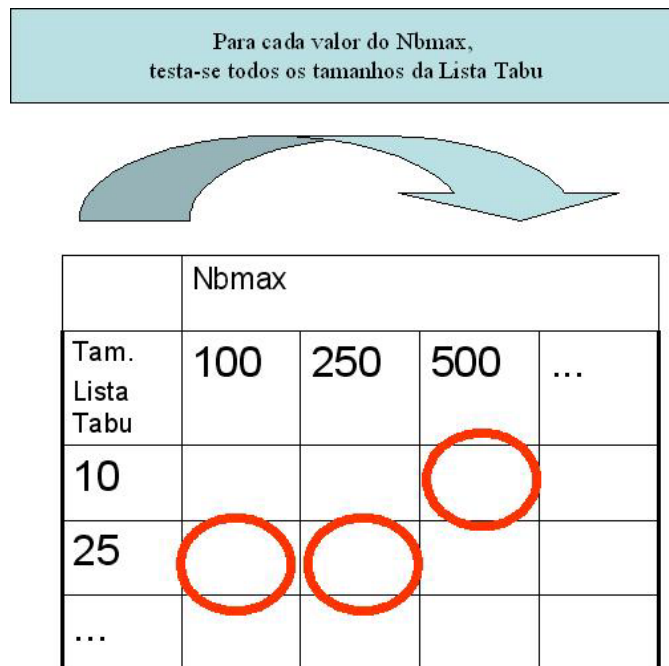
**FIGURA 22b** - Possíveis melhores resultados analisados sob perspectiva 1.

A figura 23a ilustra a relação entre o tamanho da Lista Tabu e Nbmax existente na perspectiva 2:



**FIGURA 23a** - Variação do valor do tamanho da Lista Tabu para o Nbmax.

Na figura 23a é possível observar que cada valor do Nbmax é utilizado com todos os tamanhos de Lista Tabu. Como melhor resultado é considerado aquele que apresentar o menor custo entre todos os resultados gerados para um valor fixo do Nbmax e os tamanhos diversos da Lista Tabu. A figura 23b a seguir, ilustra, sob a perspectiva 2, a localização de possíveis melhores resultados. Conforme pode ser visto o melhor resultado é aquele que apresentar o menor custo entre todos os resultados gerados para aquele valor de Nbmax e variação proposta do tamanho da Lista Tabu. Desta maneira pode haver mais de um melhor resultado para o mesmo tamanho de Lista Tabu, entretanto para cada Nbmax, apenas o resultado de menor custo é considerado o melhor. Sendo assim, de acordo com os valores propostos para o Nbmax é possível gerar 9 melhores resultados para cada conjunto de experimentos.



**FIGURA 23b** - Possíveis melhores resultados analisados sob perspectiva 2.

Para a geração dos experimentos foi definido um tamanho de 100 posições para a Lista de Soluções Elite e que a estratégia de intensificação será aplicada a cada 15 iterações sem melhora na função objetivo. Quando a estratégia de intensificação for executada, o processo de busca visita todas as soluções que estão na Lista de Soluções Elite gerando os movimentos V1' e/ou V2' para todas as soluções da lista. Os experimentos serão rodados em um microcomputador Pentium(R) 4, com 1GB de memória RAM e CPU de 3.00GHz.

## 6.2 RESULTADOS OBTIDOS

A partir dos experimentos evidenciou-se, no processo de geração de vizinhanças, que a obtenção de uma grande quantidade de soluções melhora a qualidade da busca no espaço de pesquisa (vizinhança). A flexibilidade da Busca Tabu permite incorporar diversos mecanismos de geração de vizinhança que podem ser usados para tornar a busca mais robusta. Nesta aplicação,

uma estratégia de intensificação foi usada com o intuito de aumentar e diversificar as vizinhanças geradas. Análises comparativas entre o processo de busca com e sem o uso da estratégia de intensificação foram geradas com o propósito de avaliar a qualidade dos processos de busca. Os experimentos foram realizados para os problemas 1, 2, 3, 4 e 5 de Christofides, Minggozi e Toth (1979), com 50, 75, 100, 150 e 199 clientes, respectivamente, além de 1 depósito. Estes problemas fazem parte de um conjunto de 14 problemas freqüentemente utilizados para testes na literatura (Rego, 1998; Taillard, 1993; Barbarasoglu e Ozgur, 1999; Gendreau et al, 1994; Xu e Kelly, 1996; Toth e Vigo, 2003). Para cada um dos problemas, os experimentos foram realizados variando-se os valores do Nbmax, da Lista Tabu e os movimentos geradores de vizinhança conforme valores descritos na seção 7.1. Os experimentos foram divididos em duas etapas, na primeira etapa, foram realizados 162 experimentos sem o uso da estratégia de intensificação e na segunda etapa, os experimentos foram realizados com o uso da estratégia de intensificação. A seguir serão apresentados os resultados obtidos.

### **6.2.1 Análise sob Perspectiva 1 – Variação do Nbmax para cada tamanho da Lista Tabu**

Neste item serão apresentados os resultados relacionados à análise feita sob a perspectiva 1, que se refere a variação do Nbmax para cada valor da Lista Tabu. As figuras 27 a 36 ilustram o comportamento do modelo, para algumas instâncias selecionadas, frente a variações do número máximo de iterações sem melhora da Busca Tabu e do tamanho da Lista da Tabu considerando os movimentos geradores de vizinhança.

Para facilitar a leitura, adota-se as seguintes nomenclaturas: “média inferior” para a média dos resultados gerados com Nbmax = 100 e Nbmax = 250 e “média superior” para a média dos resultados gerados com Nbmax = 1750 e Nbmax = 2000.

A figura 24 a seguir, representa os experimentos gerados com o movimento V1 para o problema 1 sem uso da intensificação no processo de busca. A média inferior foi 685,54 e a média superior 636,30.

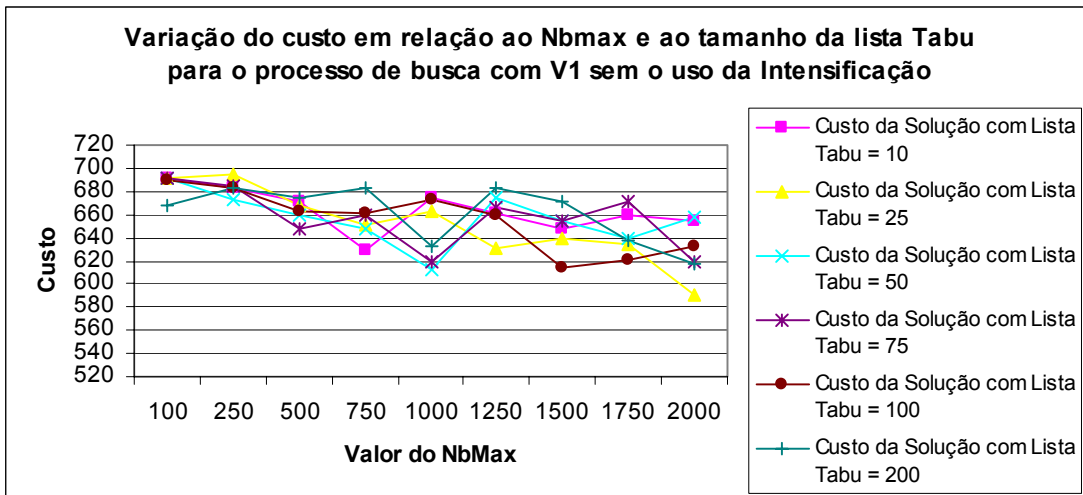


FIGURA 24 - Variação do Custo para o problema 1 usando V1 sem intensificação.

A figura 25 representa os experimentos gerados com o movimento V1 utilizando a estratégia de busca com intensificação. Neste caso, a média inferior foi 677,27 e a média superior foi 623,22.

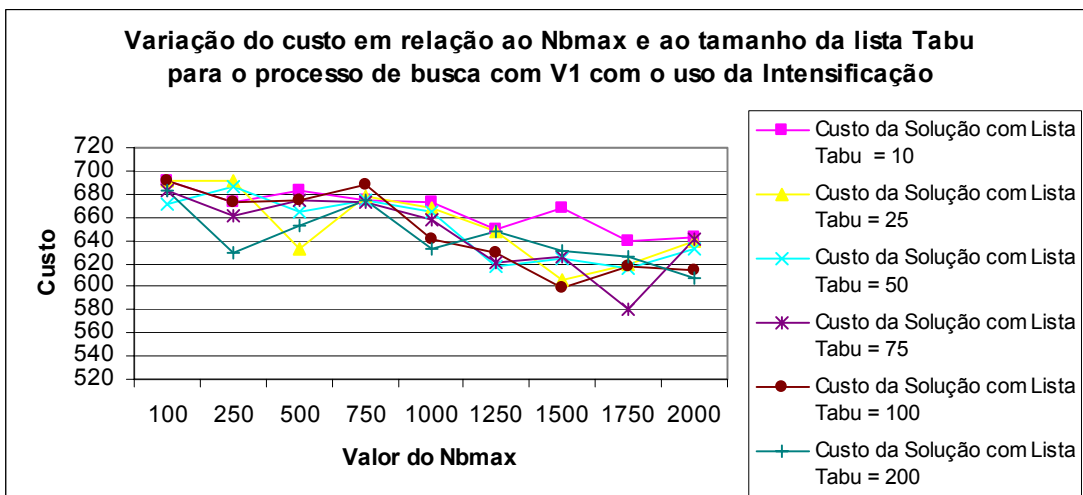
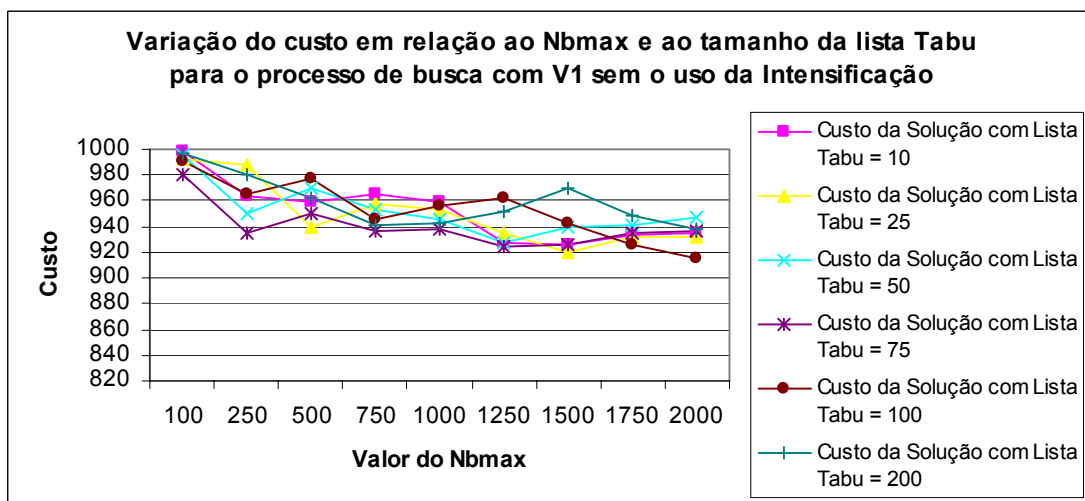


FIGURA 25 - Variação do Custo para o problema 1 usando V1 com intensificação.



As figuras de 26 e 27 a seguir representam os experimentos para o problema 2. A figura 29 representa os experimentos gerados com o movimento V1 sem uso da estratégia de intensificação no processo de busca. A média inferior foi 978,20 e a média superior 934,86.



**FIGURA 26** - Variação do Custo para o problema 2 usando V1 sem intensificação.

A figura 27 representa os experimentos gerados com o movimento V1 utilizando a estratégia de intensificação. Neste caso, a média inferior foi 978,85 e a média superior foi 921,06.

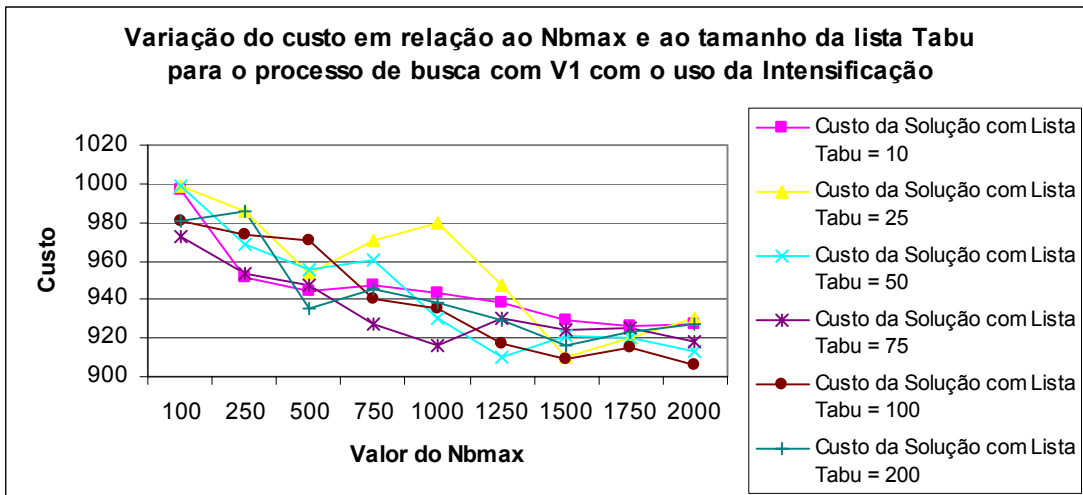


FIGURA 27 - Variação do Custo para o problema 2 usando V1 com intensificação.

As figuras de 28 e 29 representam os experimentos para o problema 3. A figura 28 representa os experimentos gerados com o movimento V2 sem o uso da estratégia de intensificação. Neste caso, a média inferior foi 922,48 e a média superior foi 887,94.

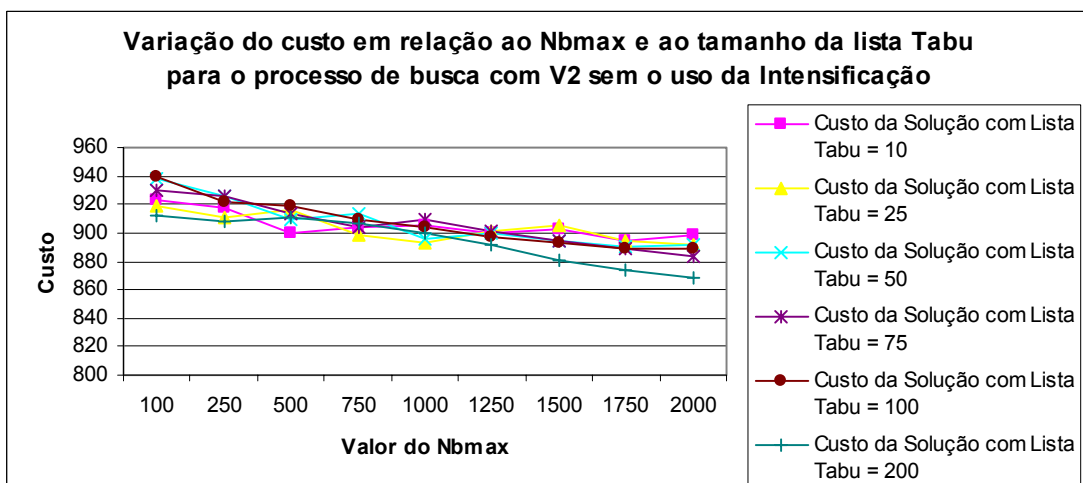
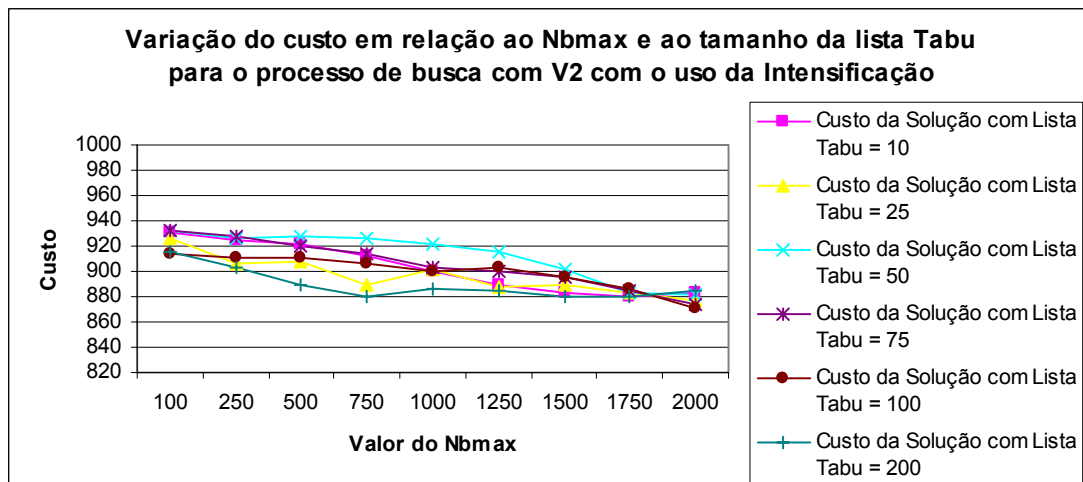


FIGURA 28 - Variação do Custo para o problema 3 usando V2 com intensificação.

A figura 29 representa os experimentos gerados com o movimento V2 utilizando a estratégia de intensificação. Neste caso, a média inferior foi 920,73 e a média superior foi 880,68.



**FIGURA 29** - Variação do Custo para o problema 3 usando V2 com intensificação.

As figuras de 30 e 31 representam os experimentos para o problema 3. A figura 30 representa os experimentos gerados com o movimento para V1 sem o uso da estratégia de intensificação. Neste caso, a média inferior foi 1228,03 e a média superior foi 1204,75.

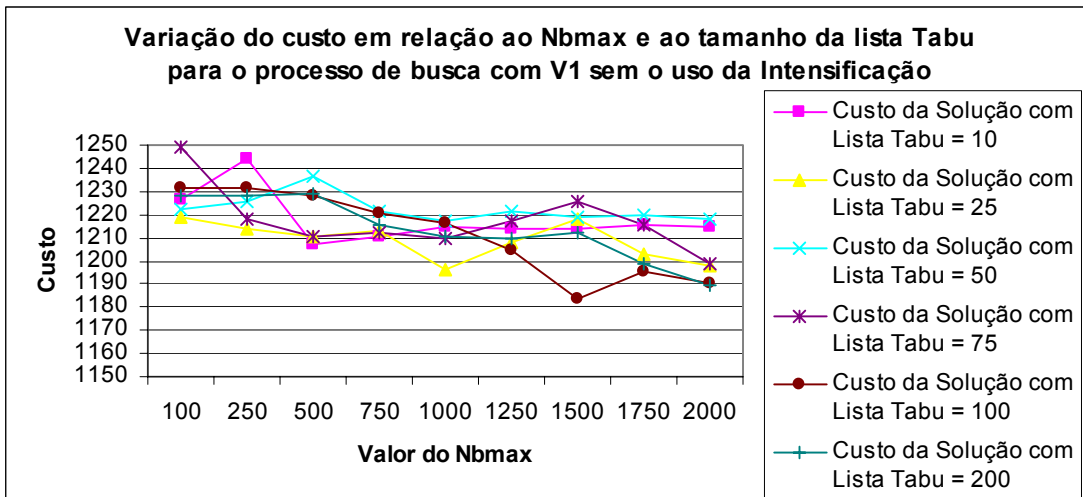


FIGURA 30 - Variação do Custo para o problema 4 usando V1 sem intensificação.

A figura 31 representa os experimentos gerados para V2 com intensificação. Neste caso, a média inferior foi 1136,01 e a média superior foi 1109,22.

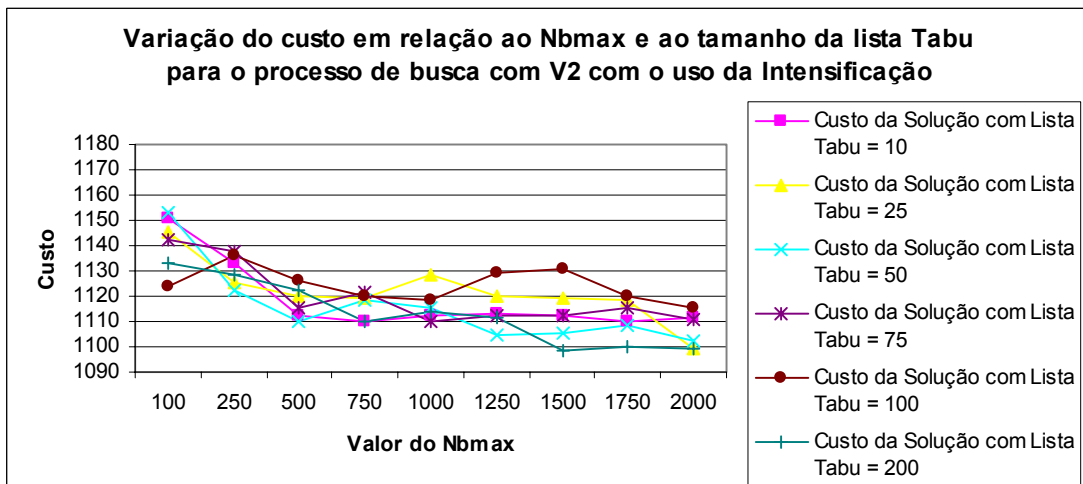
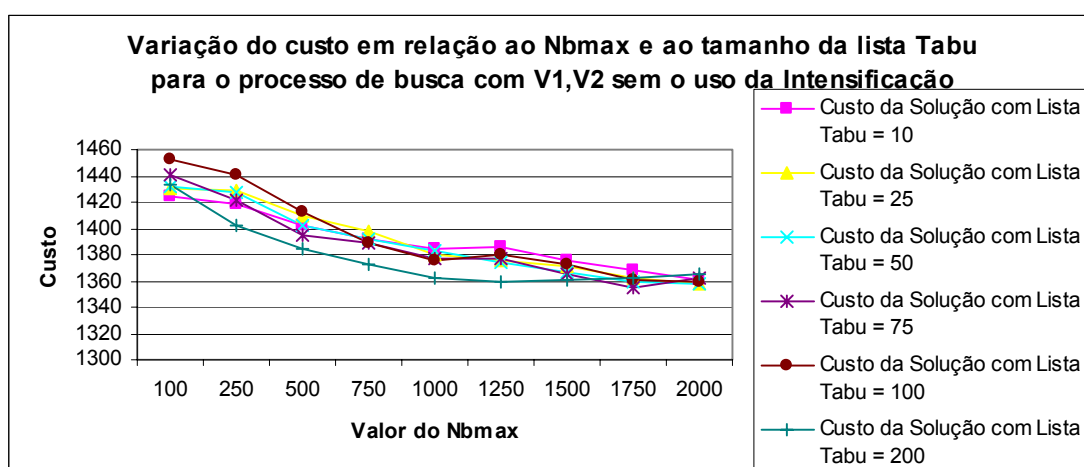


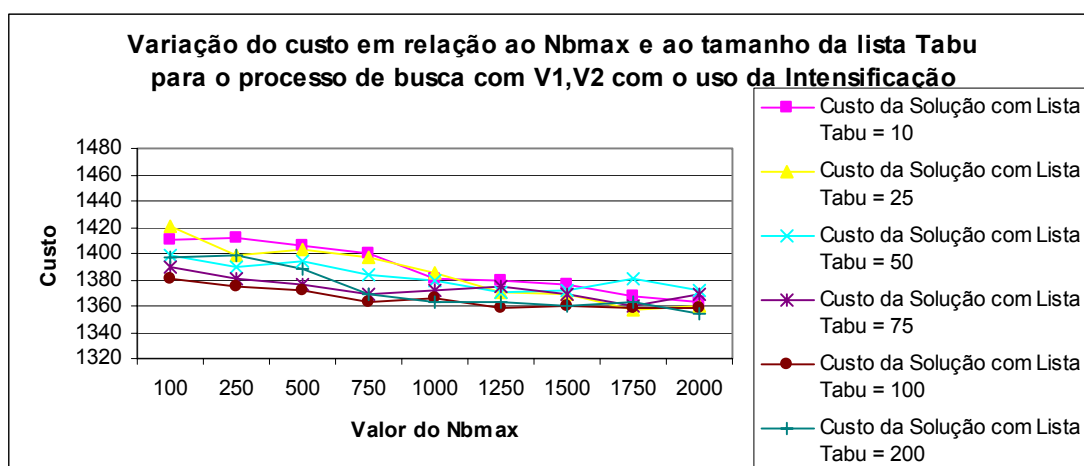
FIGURA 31 - Variação do Custo para o problema 4 usando V2 com intensificação.

A figura 32 representa os experimentos gerados com os movimentos V1,V2 sem o uso da estratégia de intensificação. Neste caso, a média inferior foi 1429,67 e a média superior foi 1261,10.



**FIGURA 32** - Variação do Custo para o problema 5 usando V1,V2 sem intensificação.

A figura 33 representa os experimentos gerados com os movimentos V1,V2 utilizando a estratégia de intensificação no processo de busca. Neste caso, a média inferior foi 1395,81 e a média superior foi 1363,53.



**FIGURA 33** - Variação do Custo para o problema 5 usando V1,V2 com intensificação.

Através das figuras é possível visualizar que à medida que o valor do Nbmax aumenta, mais soluções de menor custo são geradas. A tabela 5 a seguir apresenta os resultados da média inferior e superior para todos os experimentos realizados.

**TABELA 5** – Resultados das Médias Superiores e Inferiores.

	Movimento	Média Superior	Média Inferior
Problema 1	V1	685,54	636,30
	V1 + V1'	677,27	623,22
	V2	604,60	571,69
	V2 + V2'	596,89	554,79
	V1,V2	546,98	532,25
	V1,V2 + V1',V2'	557,45	531,58

	Movimento	Média Superior	Média Inferior
Problema 2	V1	978,20	934,86
	V1 + V1'	978,85	921,06
	V2	921,40	879,44
	V2 + V2'	907,92	868,22
	V1,V2	886,74	859,16
	V1,V2 + V1',V2'	884,63	850,19
Problema 3	V1	971,30	937,79
	V1 + V1'	970,34	936,12
	V2	922,48	887,94
	V2 + V2'	920,73	880,68
	V1,V2	895,64	862,44
	V1,V2 + V1',V2'	886,77	853,99
Problema 4	V1	1228,03	1204,75
	V1 + V1'	1216,33	1201,84
	V2	1144,62	1115,03
	V2 + V2'	1136,01	1090,22
	V1,V2	1106,67	1075,57
	V1,V2 + V1',V2'	1098,63	1070,85
Problema 5	V1	1578,93	1561,63
	V1 + V1'	1573,17	1550,75
	V2	1451,13	1403,00
	V2 + V2'	1384,43	1363,13

	Movimento	Média Superior	Média Inferior
	V1,V2	1429,17	1361,10
	V1,V2 + V1',V2'	1395,12	1363,53

Analisando a tabela nota-se que em 100% dos experimentos realizados o valor da média superior é menor do que o valor da média inferior. Este fato indica que o aumento do Nbmax tem como consequência a geração de soluções de menor custo. Observou-se nos experimentos que durante o processo de busca, 94,44% dos melhores resultados do problema 1 foram obtidos com Nbmax maior que 1000. Para o problema 2 este percentual foi de 97,22% e para o problema 3, 94,44%. Para os problemas 4 e 5 este percentual foi de 100%. Estas observações mostram que um grande percentual de melhores soluções é gerado para valores altos do Nbmax, indicando que a qualidade das soluções é aumentada ao se usar valores altos para o Nbmax.

As tabelas 6 a 10 indicam o valor de Nbmax e a quantidade de melhores resultados que foram encontrados naquele Nbmax sob a perspectiva 1. Nas tabelas a seguir os dados aparecem agrupados por problema e classificados de acordo com o processo de busca utilizado (com intensificação ou sem intensificação), sem diferenciação quanto aos movimentos utilizados. Cada conjunto gera 6 melhores resultados (um resultado para cada tamanho da lista), desta maneira a seguir são apresentadas a localização dos 36 melhores resultados de cada problema. Os valores de Nbmax que não aparecem nas tabelas ou aparecem com valor 0 (zero) não tiveram nenhum melhor resultado gerado para aquele valor de Nbmax.



**TABELA 6** - Quantidade de melhores resultados encontrados para variação proposta do Nbmax aplicado ao problema 1.

Tipo\Nbmax	500	750	1000	1250	1500	1750	2000
Sem uso da Intensificação	1	1	3	3	3	4	3
Com uso da Intensificação	0	0	1	2	4	6	5

**TABELA 7** - Quantidade de melhores resultados encontrados para variação proposta do Nbmax aplicado ao problema 2.

Tipo\Nbmax	500	1000	1250	1500	1750	2000
Sem uso da Intensificação	1	1	3	4	2	7
Com uso da Intensificação	0	1	2	3	6	6

**TABELA 8** - Quantidade de melhores resultados encontrados para variação proposta do Nbmax aplicado ao problema 3.

Tipo\Nbmax	750	1250	1500	1750	2000
Sem uso da Intensificação	1	2	1	4	10
Com uso da Intensificação	1	0	2	1	12

**TABELA 9** - Quantidade de melhores resultados encontrados para variação proposta do Nbmax aplicado ao problema 4.

Tipo\Nbmax	1000	1250	1500	1750	2000
Sem uso da Intensificação	1	2	3	1	11
Com uso da Intensificação	1	4	1	2	10

**TABELA 10** - Quantidade de melhores resultados encontrados para variação proposta do Nbmax aplicado ao problema 5.

Tipo\Nbmax	1000	1250	1500	1750	2000
Sem uso da Intensificação	2	3	0	1	12
Com uso da Intensificação	0	1	1	5	11

A análise dos resultados apresentados nas tabelas 6 a 10 mostra que os valores altos de Nbmax foram os que apresentaram o maior número de melhores resultados, indicando que valores altos de Nbmax contribuem para o aumento da qualidade do processo de busca.

### 6.2.2 Análise sob à Perspectiva 2 - Variação do tamanho da Lista Tabu para Nbmax

Neste item serão apresentados os resultados relacionados à análise feita sob a perspectiva 2, que se refere a variação do tamanho da Lista Tabu para cada valor de Nbmax. Para o problema 1, sem intensificação, 66,66% dos melhores resultados foram obtidos com tamanho da lista  $\geq 75$  e com intensificação este percentual foi de 48,14%. Para o problema 2, os percentuais foram 55,55% e 59,25% sem intensificação e com intensificação respectivamente. Para o problema 3 os percentuais foram 77,77% sem intensificação e 96,29% com intensificação. Para o problema 4 os percentuais foram de 74,07% e 77,77%, sem intensificação e com intensificação,

respectivamente. Para o problema 5 os percentuais foram, 63,88% sem intensificação e 92,59% com intensificação. As tabelas 11 a 15 apresentam os números indicando a quantidade de melhores resultados obtidos para os tamanhos de Lista Tabu maior e menor que 75.

**TABELA 11** - Quantidade de melhores resultados para tamanho de lista menor e maior que 75 para problema 1.

Movimento\Tipo	Sem intensificação		Com intensificação	
	<75	>=75	<75	>=75
V1	4	5	3	6
V2	3	6	7	2
V1V2	2	7	4	5

**TABELA 12** - Quantidade de melhores resultados para tamanho de lista menor e maior que 75 para problema 2.

Movimento\Tipo	Sem intensificação		Com intensificação	
	<75	>=75	<75	>=75
V1	2	7	3	6
V2	6	3	4	5
V1V2	4	5	4	5

**TABELA 13** - Quantidade de melhores resultados para tamanho de lista menor e maior que 75 para problema 3.

Movimento\Tipo	Sem intensificação		Com intensificação	
	<75	>=75	<75	>=75
V1	3	6	1	8
V2	3	6	0	9
V1V2	0	9	0	9

**TABELA 14** - Quantidade de melhores resultados para tamanho de lista menor e maior que 75 para problema 4.

Movimento\Tipo	Sem intensificação		Com intensificação	
	<75	>=75	<75	>=75
V1	4	5	3	6
V2	3	6	2	7
V1V2	0	9	1	8

**TABELA 15** - Quantidade de melhores resultados para tamanho de lista menor e maior que 75 para problema 5.

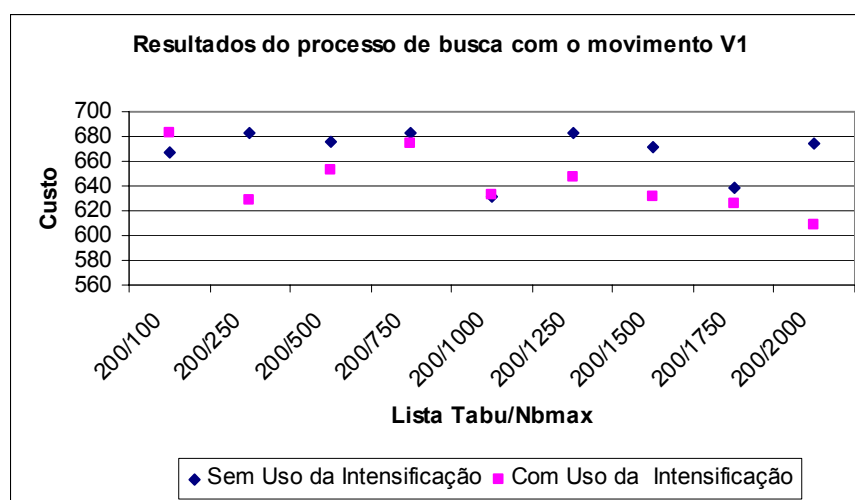
Movimento\Tipo	Sem intensificação		Com intensificação	
	<75	>=75	<75	>=75
V1	3	6	1	8
V2	0	9	1	8
V1V2	1	8	0	9

A partir dessas informações notamos que tamanhos grandes ( $\geq 75$ ) para a Lista Tabu aumentam em, pelo menos, 40% a qualidade do processo de busca. A análise destas tabelas mostra que tamanhos grandes para a Lista Tabu contribuem para a geração de soluções de menor custo, aumentando a qualidade do processo de busca.

### **6.2.3 Análise Comparativa do processo de Busca com Uso da Estratégia de Intensificação e sem o Uso da Estratégia de Intensificação**

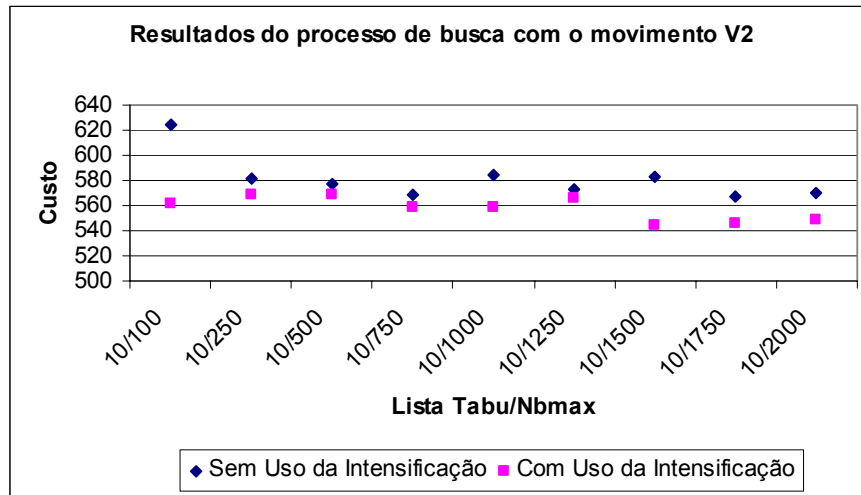
A análise a seguir tem por objetivo comparar o processo de busca com e sem o uso da estratégia de intensificação além de permitir uma análise da qualidade dos movimentos geradores de vizinhança através da comparação dos diferentes resultados obtidos para instâncias iguais. As figuras 37 a 46 ilustram os resultados de alguns experimentos permitindo a comparação da qualidade do processo de busca com e sem o uso da estratégia de intensificação. Os pontos nos gráficos representam os resultados obtidos para os valores indicados para a Lista Tabu e o

Nbmax. Analisando a figura 34, que representa o processo de busca para o problema 1, com Lista Tabu = 200 e usando o movimento V1, observa-se que o uso da estratégia de intensificação melhorou o custo de 77,77% dos resultados.



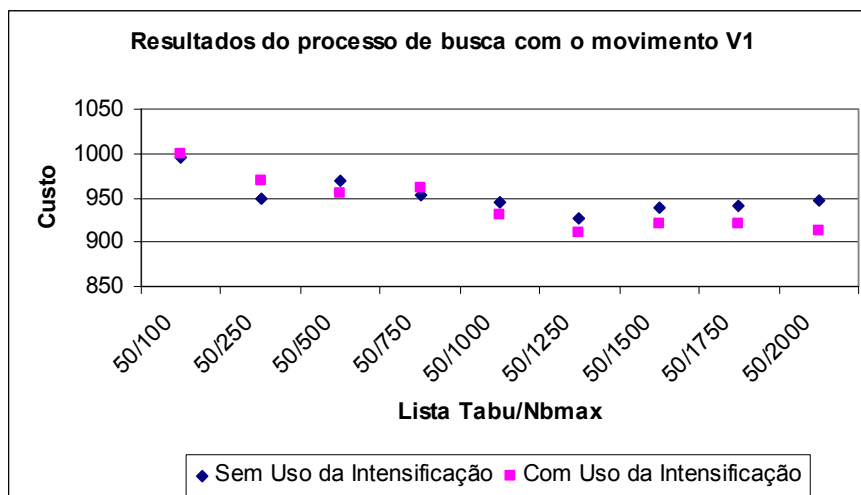
**FIGURA 34** - Processo de busca usando V1 e Lista Tabu = 200 para o problema 1.

A figura 35, que representa o processo de busca para o problema 1, com Lista Tabu = 10 e usando o movimento V2, mostra que estratégia de intensificação melhorou 100% dos resultados.



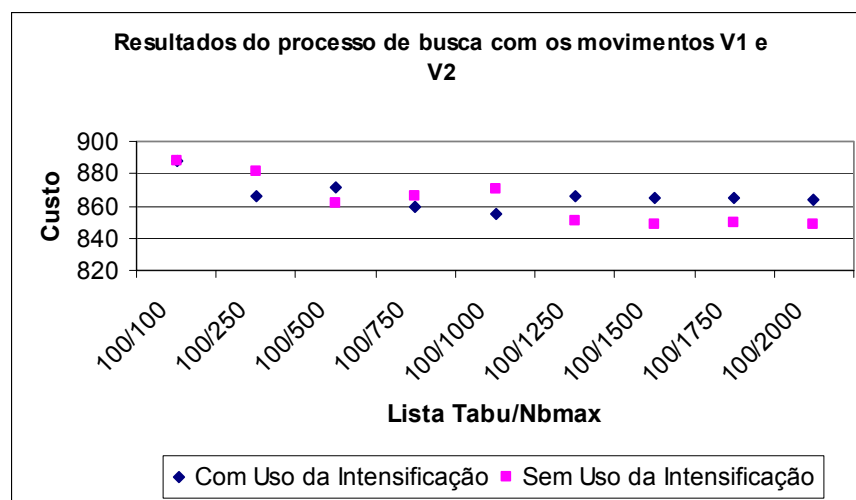
**FIGURA 35** - Processo de busca usando V2 e Lista Tabu = 10 para o problema 1.

A figura 36, que representa o processo de busca para o problema 2, com Lista Tabu = 50 e usando o movimento V1, mostra que a estratégia de intensificação melhorou 66,66% dos resultados.



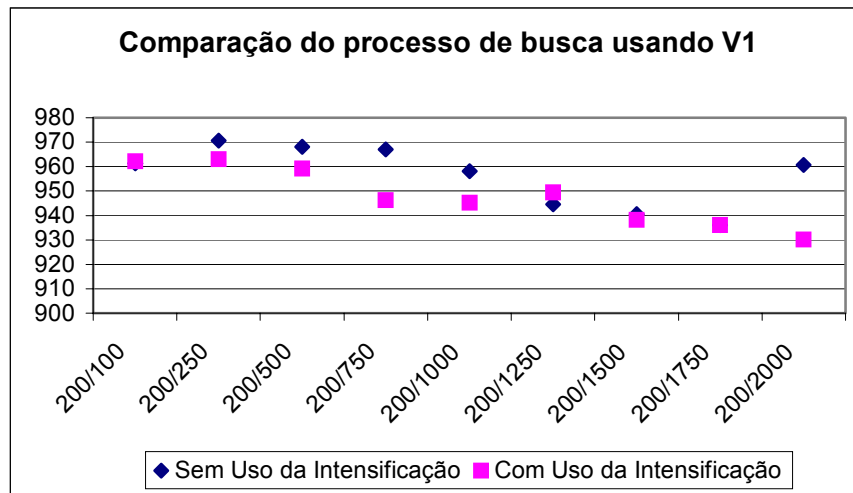
**FIGURA 36** - Processo de busca usando V1 e Lista Tabu = 50 para o problema 2.

A figura 37, que representa o processo de busca para o problema 2, com Lista Tabu = 100 e usando os movimentos V1 e V2 juntos, mostra que a estratégia de intensificação melhorou 55,55% dos resultados.



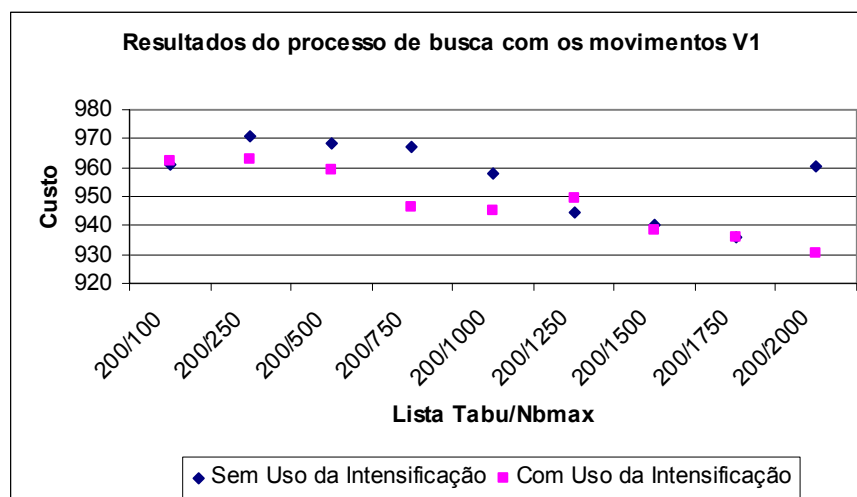
**FIGURA 37** - Processo de busca usando V1, V2 e Lista Tabu = 100 para o problema 2.

A figura 38, que representa o processo de busca para o problema 3, com Lista Tabu = 200 e usando o movimento V1, mostra que 66,66% dos resultados foram melhorados.



**FIGURA 38** - Processo de busca usando V1 e Lista Tabu = 200 para o problema 3.

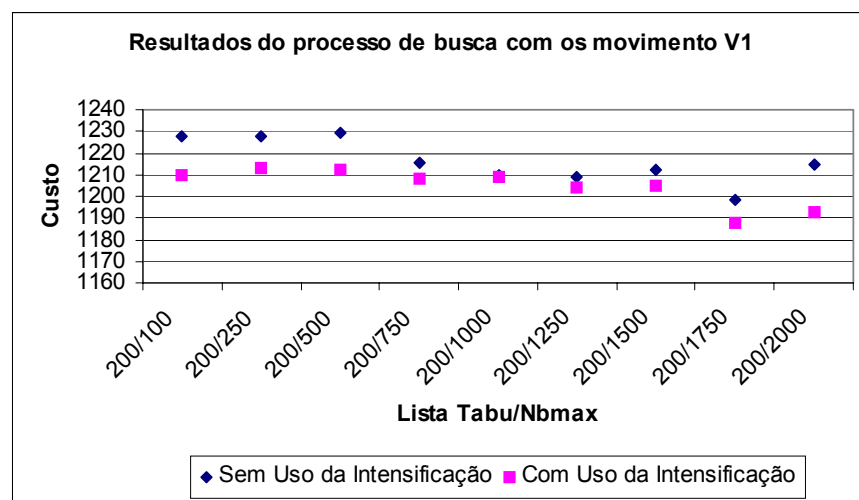
A figura 39, que representa o processo de busca para o problema 3, com Lista Tabu = 200 e usando o movimento V1, mostra que a estratégia de intensificação melhorou 66,66% dos resultados.



**FIGURA 39** - Processo de busca usando V1 e Lista Tabu = 200 para o problema 3.

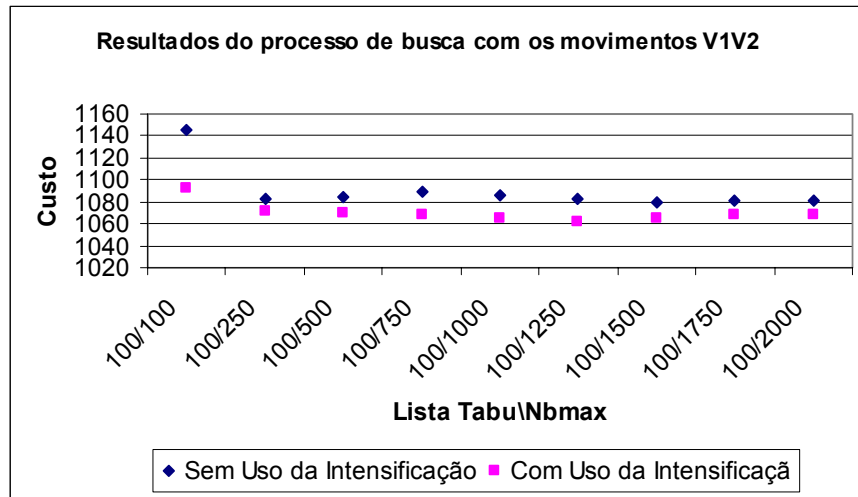


A figura 40, que representa o processo de busca para o problema 3, com Lista Tabu = 200 e usando o movimento V1, mostra que a estratégia de intensificação melhorou 88,88% dos resultados.



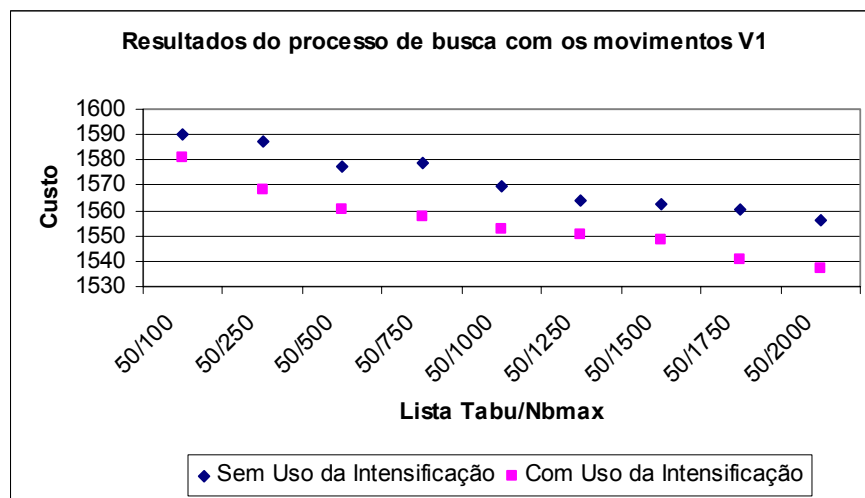
**FIGURA 40** - Processo de busca usando V2 e Lista Tabu = 200 para o problema 4.

A figura 41, que representa o processo de busca para o problema 3, com Lista Tabu = 100 e usando os movimento V1 e V2 juntos, mostra que a estratégia de intensificação melhorou 100% dos resultados.



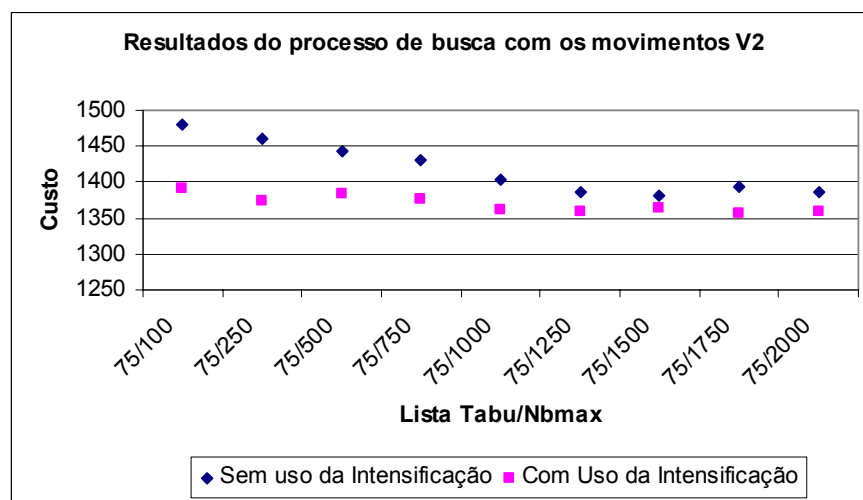
**FIGURA 41** - Processo de busca usando V1,V2 e Lista Tabu = 200 para o problema 4.

A figura 42, que representa o processo de busca para o problema 3, com Lista Tabu = 50 e usando os movimentos V1 e V2 juntos, mostra que a estratégia de intensificação melhorou 100% dos resultados.



**FIGURA 42** - Processo de busca usando V1 e Lista Tabu = 50 para o problema 5.

A figura 43, que representa o processo de busca para o problema 5, com Lista Tabu = 75 e usando os movimentos V1 e V2 juntos, mostra que a estratégia de intensificação melhorou 100% dos resultados.



**FIGURA 43** - Processo de busca usando V2 e Lista Tabu = 200 para o problema 5.

Através dos gráficos apresentados, observa-se a diferença na qualidade dos resultados gerados pelo processo de busca com o uso da estratégia de intensificação. A estratégia de intensificação aplicada neste trabalho utiliza os mesmos movimentos executados no processo de busca sem intensificação, entretanto, aumentando o tamanho e a diversidade das soluções geradas que irão compor a vizinhança de uma solução.

Os gráficos das figuras 44 à 48 a seguir mostram a quantidade de resultados que apresentaram melhora com o uso da estratégia de intensificação. Para o problema 1, dos 162 resultados gerados para os experimentos sem intensificação, a estratégia de intensificação melhorou a qualidade de 97 deles. Para o problema 2, 121 resultados foram melhorados. Para o problema 3, 102 resultados foram melhorados com o uso da estratégia de intensificação. Para o problema 4, 117 foram melhorados e para o problema 5, a estratégia de intensificação melhorou a qualidade de 135.

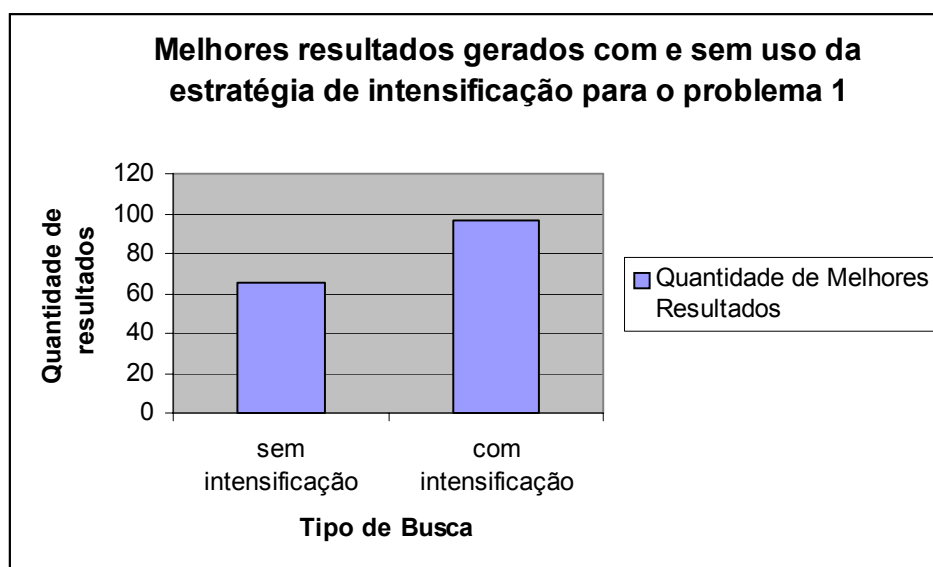


FIGURA 44 - Quantidade de resultados gerados com e sem intensificação para o problema 1.

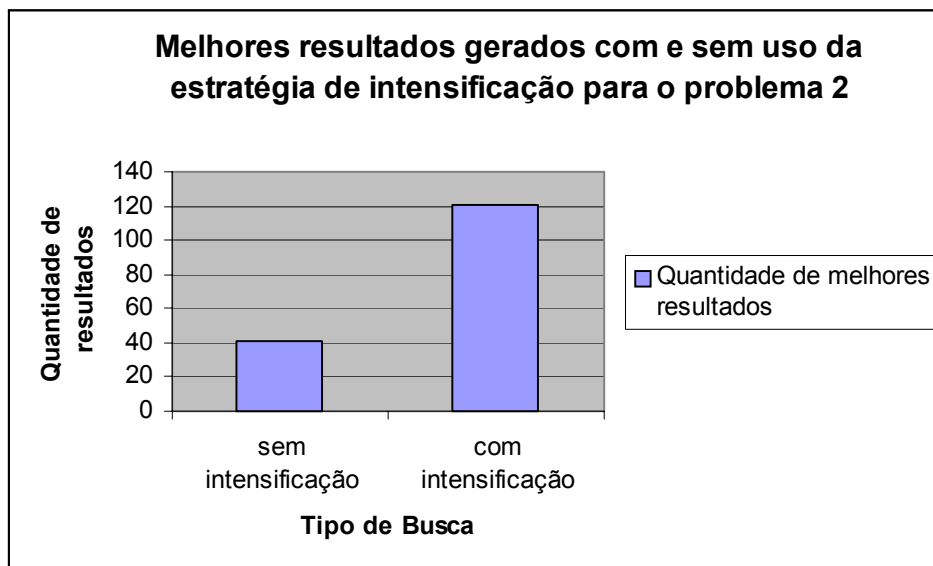
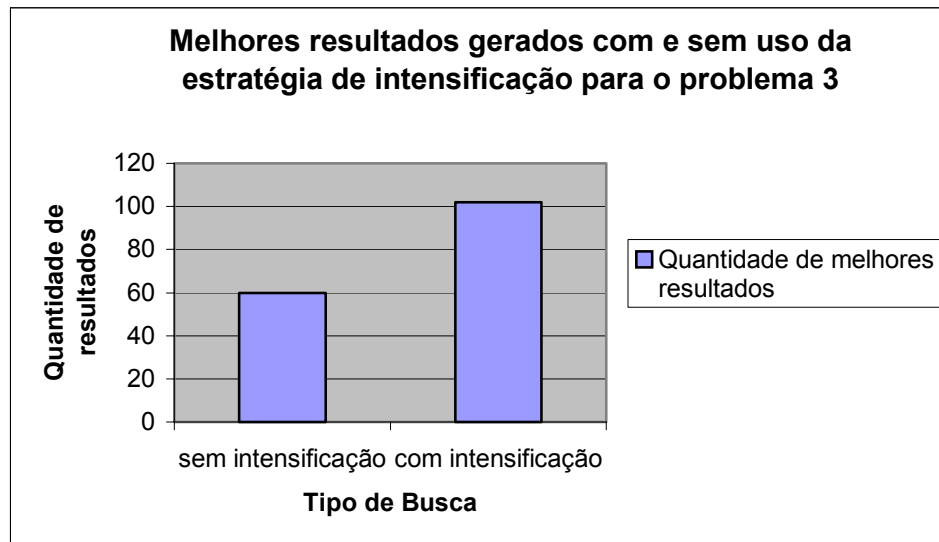
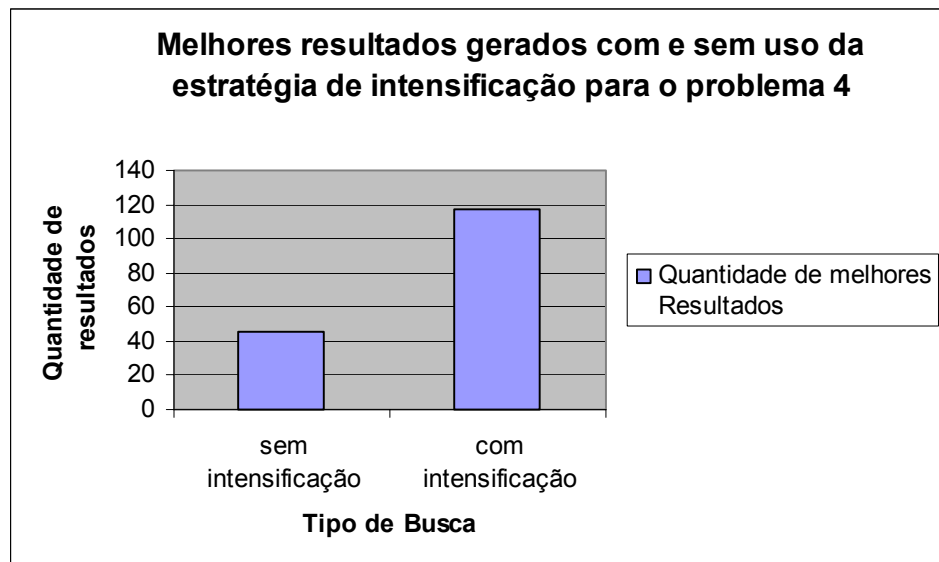


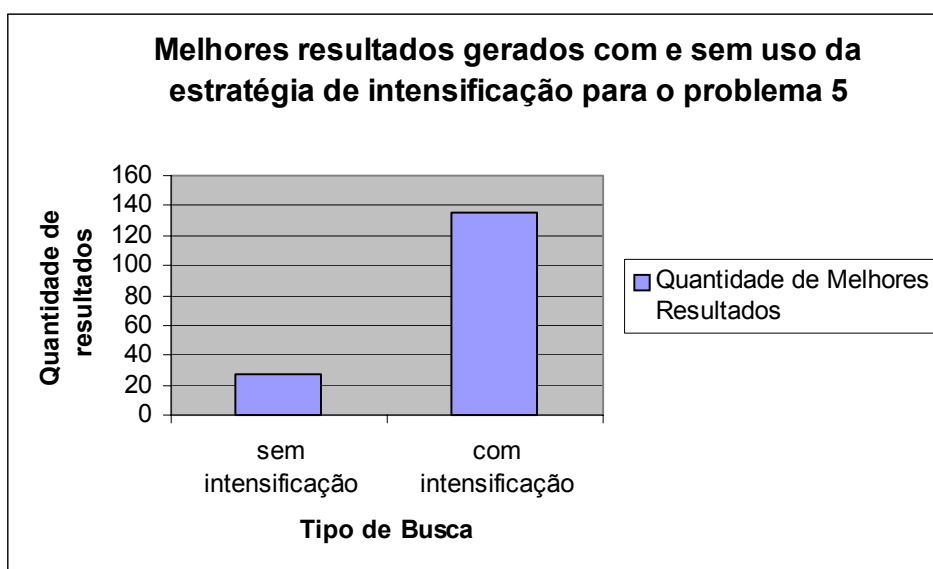
FIGURA 45 - Quantidade de resultados gerados com e sem intensificação para o problema 2.



**FIGURA 46** - Quantidade de resultados gerados com e sem intensificação para o problema 3.



**FIGURA 47** - Quantidade de resultados gerados com e sem intensificação para o problema 4.



**FIGURA 48** - Quantidade de resultados gerados com e sem intensificação para o problema 5.

Os gráficos mostram que para todos os problemas, o uso da estratégia de intensificação apresentou um aumento na qualidade bem significativo. A qualidade dos resultados sofreu melhora de, no mínimo 50%.

A análise a seguir visa comparar os resultados obtidos para os mesmos valores de  $N_{bmax}$  e tamanho da Lista Tabu nos dois processos de busca (sem intensificação e com intensificação) considerando os diferentes movimentos utilizados. As tabelas 16 a 20 apresentam o número de melhores resultados obtidos com e sem o uso da estratégia de intensificação nos experimentos. A coluna 1 apresenta o movimento utilizado. A segunda coluna apresenta a quantidade de resultados cujo valor foi melhor na busca sem intensificação e a terceira coluna apresenta a quantidade de resultados que a estratégia de intensificação conseguiu melhorar.

**TABELA 16** - Quantidade de melhores resultados encontrados para o problema 1.

Mov.\Tipo	Sem Intensificação	Com Intensificação
V1	22	32
V2	12	42
V1V2	31	23

**TABELA 17** - Quantidade de melhores resultados encontrados para o problema 2.

4Mov.\Tipo	Sem Intensificação	Com Intensificação
V1	14	40
V2	9	45
V1V2	18	36

**TABELA 18** - Quantidade de melhores resultados encontrados para o problema 3.

Mov.\Tipo	Sem Intensificação	Com Intensificação
V1	21	33
V2	21	33
V1V2	18	36

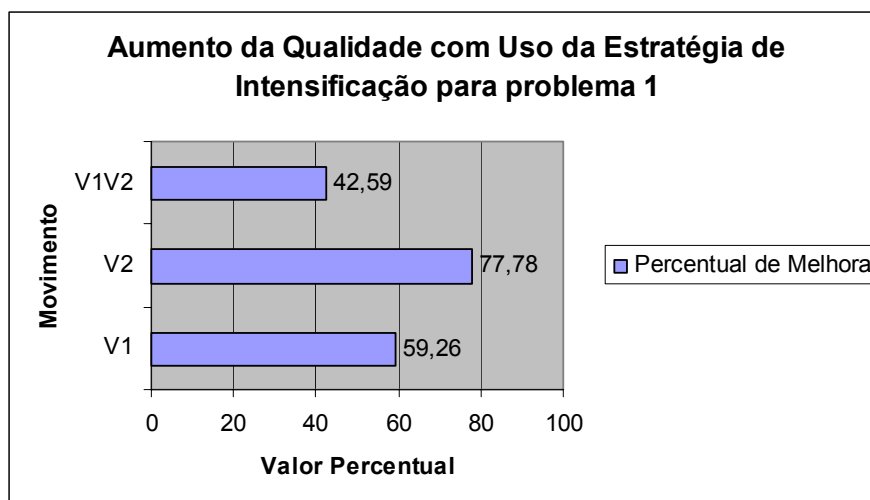
**TABELA 19** - Quantidade de melhores resultados encontrados para o problema 4.

Mov.\Tipo	Sem Intensificação	Com Intensificação
V1	16	38
V2	11	43
V1V2	16	38

**TABELA 20**- Quantidade de melhores resultados encontrados para o problema 5.

Mov.\Tipo	Sem Intensificação	Com Intensificação
V1	11	43
V2	0	54
V1V2	16	38

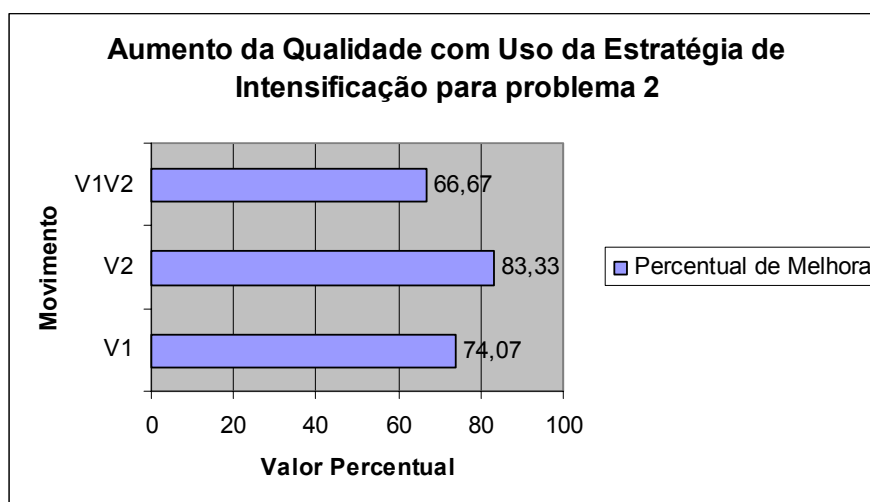
Os gráficos a seguir ilustram o percentual de melhora dos resultados obtidos através do uso da busca com intensificação com relação e busca sem intensificação. São ilustrados 15 casos correspondendo aos 3 movimentos utilizados para cada um dos 5 problemas. Para o problema 1, usando V1, V2 e V1,V2 juntos, os percentuais de melhora com o uso da estratégia de intensificação foram respectivamente 59,26%, 77,78% e 42,59%. Estes percentuais são ilustrados no gráfico da figura 49.



**FIGURA 49** - Melhoria gerada em função do uso da estratégia de intensificação para o problema 1.

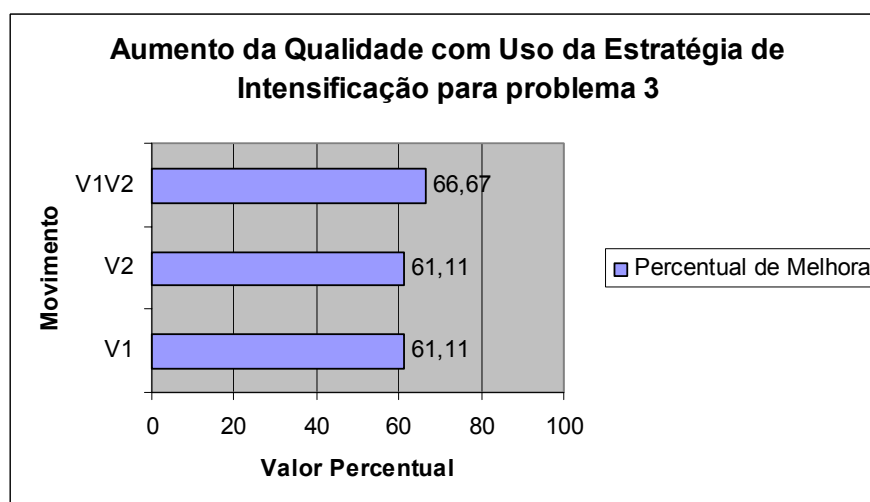
Para o problema 2, usando V1, V2 e V1,V2 juntos, os percentuais de melhora com o uso da estratégia de intensificação foram respectivamente 74,07%, 83,33% e 66,67%. Estes percentuais são ilustrados no gráfico da figura 50.





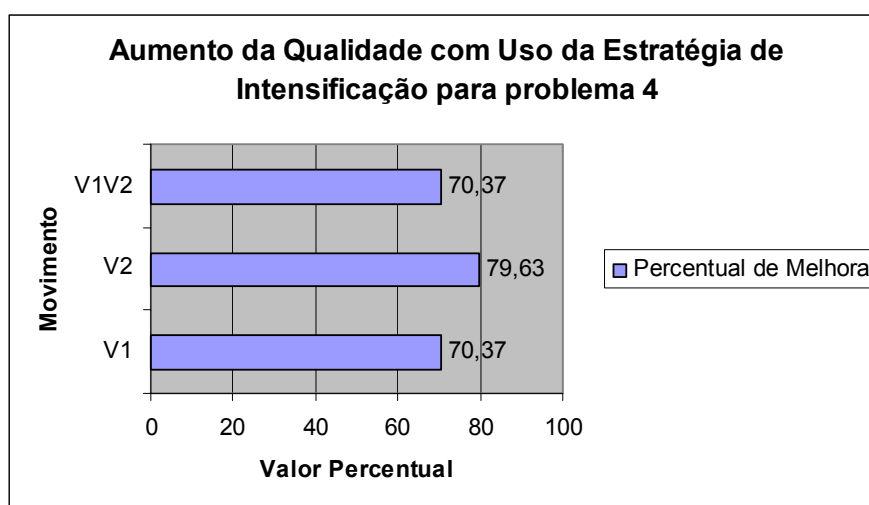
**FIGURA 50** - Melhoria gerada em função do uso da estratégia de intensificação para o problema 2.

Para o problema 3, usando V1, V2 e V1,V2 juntos, os percentuais de melhora com o uso da estratégia de intensificação foram respectivamente 61,11%, 61,11% e 66,67%. Estes percentuais são ilustrados no gráfico apresentados na figura 51.



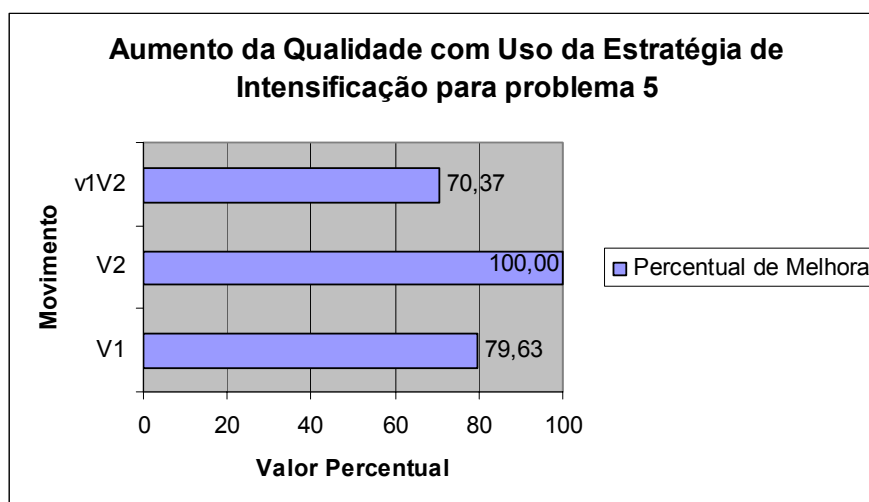
**FIGURA 51** - Melhoria gerada em função do uso da estratégia de intensificação para o problema 3.

Para o problema 4, usando V1, V2 e V1,V2 juntos, os percentuais de melhora com o uso da estratégia de intensificação foram respectivamente 70,37%, 79,63% e 70,37%. Estes percentuais são ilustrados no gráfico apresentados na figura 52.



**FIGURA 52** - Melhoria gerada em função do uso da estratégia de intensificação para o problema 4.

Para o problema 5, usando V1, V2 e V1,V2 juntos, os percentuais de melhora com o uso da estratégia de intensificação foram respectivamente 79,63%, 100% e 70,37%. Estes percentuais são ilustrados no gráfico apresentados na figura 53.



**FIGURA 53** - Melhoria gerada em função do uso da estratégia de intensificação para o problema 5.

Observando os gráficos nota-se a ocorrência de um aumento significativo na qualidade das soluções geradas pelo processo de busca com o uso da estratégia de intensificação. Em todos os casos, o aumento foi maior que 40% sendo que em 13 dos 15 casos ilustrados, o aumento foi superior a 60%. Comparando os percentuais de melhoria para os cinco problemas, nota-se que o movimento V2 foi o movimento que mais se beneficiou da estratégia de intensificação, pois apresentou o maior percentual de melhora em 4 dos 5 problemas. A tabela 21 apresenta o tamanho médio e o desvio padrão das vizinhanças geradas nos experimentos.

**TABELA 21**- Tamanho Médio e Desvio Padrão dos tamanhos de vizinhanças.

	Política	Tamanho Médio da Vizinhança	Desvio Padrão do Tamanho
Problema 1	V1	26	5,13
	V2	247	8,94
	V1,V2	298	12,00

	Política	Tamanho da Vizinhança	Desvio Padrão do Tamanho
Problema 2	V1	65	7,63
	V2	382	10,35
	V1,V2	453	13,99
Problema 3	V1	78	9,57
	V2	792	15,16
	V1,V2	889	17,19
Problema 4	V1	154	7,79
	V2	1758	15,04
	V1,V2	1890	21,48
Problema 5	V1	142	6,56
	V2	2230	26,39
	V1,V2	2310	27,57

Através da tabela 21, vê-se o tamanho médio das vizinhanças geradas pelos movimentos V1 e V2 (juntos e isolados) para cada problema. Na busca sem intensificação, V1 é o movimento que gera a menor vizinhança enquanto que a maior vizinhança é gerada quando os movimentos V1 e V2 são utilizados juntos. Ao aplicar a estratégia de intensificação o tamanho das vizinhanças aumenta. O movimento V2, que consiste na realocação de vértices foi o movimento que apresentou o aumento de qualidade mais significativo com o uso da estratégia de intensificação. Em comparação com o movimento V1, que consiste na troca de vértices, o movimento V2 é mais flexível no sentido de permitir que um número maior de vizinhos sejam gerados. No movimento V1 é necessário testar a demanda total de duas rotas, para verificar se o movimento pode ser realizado. Tanto na rota origem como na rota destino é necessário que a adição da demanda relacionada ao novo vértice que está sendo alocado na rota, não gera excesso na capacidade do veículo. Para que V1 gere um novo vizinho é necessário que haja 2 “operações de adição de demanda” (1 para cada rota envolvida) efetuadas. No movimento V2 essa verificação quanto à capacidade do veículo e demanda total da rota é realizada em apenas 1 rota (na rota que está

recebendo o vértice). Assim, apenas 1 “operação de adição de demanda” nas rotas é realizada, já que a rota que perde o seu vértice não corre o risco de exceder a capacidade do veículo. Em função da quantidade de “operações de adição de demanda” que são necessárias, nota-se que existe uma dificuldade maior em gerar vizinhos a partir do movimento V1 do que do movimento V2. Fato este que justifica o tamanho da vizinhança gerada pelo V1 ser menor do que o tamanho da vizinhança gerada por V2. Pode-se dizer que o movimento V1 está mais “engessado” e produz uma vizinhança mais pobre, pois exige que mais restrições sejam aceitas para que possa ser executado. A dinâmica do V2 permite que soluções mais diversificadas sejam geradas. Analisando o percentual de melhora obtido por V1 e por V2, conclui-se que a estratégia de intensificação é mais benéfica quando aplicada a um movimento mais flexível que consiga aproveitar dos utilizar os mecanismos da estratégia de intensificação para aumentar o tamanho da vizinhança gerada. Conforme explicado no item 6.3.3.4 a estratégia de intensificação proposta neste trabalho visa aumentar a quantidade de soluções geradas em regiões de busca historicamente boas. Aumentando o tamanho da vizinhança, aumenta-se a diversidade de soluções geradas. Se a estratégia de intensificação for aplicada a um movimento que devido a sua dinâmica não consegue gerar muitas soluções, a estratégia acaba não atingindo seu objetivo. Os experimentos indicam que em uma vizinhança grande, mais soluções de boa qualidade são geradas.

Comparando os percentuais de melhora de V2 e V1,V2 juntas, nota-se que a busca com V2 apresentou um percentual de melhora maior do que a busca realizada pelos movimentos juntos. O objetivo de usar os movimentos V1 e V2 juntos era aumentar a quantidade de vizinhos gerados e a aumentar a diversidade das soluções. Uma vez que os movimentos V1 e V2 apresentam dinâmicas diferentes para criar novas soluções, sabe-se que o uso dos dois movimentos juntos cria soluções também diferentes das que se esperaria como padrão para ser gerada por um determinado movimento. Por exemplo, em soluções geradas pelo movimento V2, existirá sempre uma rota com um acréscimo de 1 vértice enquanto que a outra sofre um decréscimo de 1 vértice. Assim o movimento V2, permite que uma rota seja excluída de uma solução através de

sucessivas realocações de seus vértices. Utilizando apenas o movimento V1, essa exclusão de rotas não é possível, uma vez que esse movimento realiza sempre uma adição nas duas rotas, não alterando a quantidade de vértices das rotas. Esta característica, além da própria dinâmica dos movimentos, aumenta a diversidade das soluções geradas, uma vez que a dinâmica de um movimento pode ser aplicada em uma solução gerada a partir do outro movimento. Como os dois movimentos geram perturbações diferentes em uma mesma solução o uso dos dois movimentos juntos, diferentes soluções surgem dessa combinação de movimentos. Assim sendo, observou-se que a qualidade das soluções geradas pelos movimentos juntos é maior do que a qualidade das soluções geradas apenas por um único movimento. Analisando os experimentos observa-se que o impacto positivo da estratégia de intensificação é maior quando a estratégia de intensificação é aplicada a movimentos flexíveis e que esse impacto se reduz se ela é aplicada a movimentos que geram resultados de boa qualidade. As tabelas 23 á 27 a seguir apresentam os resultados médios, desvios-padrão e melhores resultados obtidos para cada problema divididos por movimento utilizado e pelo processo de busca realizado. Para os cinco problemas testados, um mesmo padrão de qualidade relacionado aos movimentos geradores dos resultados foi observado. Nota-se que os melhores resultados e as menores médias foram obtidas com o uso dos movimentos agrupados. Neste caso, dois aspectos são usados como justificativa: com o uso dos movimentos agrupados, o tamanho da vizinhança gerada é maior assim como a diversidade de soluções também. O movimento que produziu os piores resultados, com as maiores médias foi o movimento V1. O que não se considera um fato anormal uma vez que em discussão anterior já é foi constatado que o movimento V1 o mais “engessado” e o responsável por produzir uma vizinhança mais pobre. Todos os melhores resultados foram obtidos com Lista Tabu = 200, Nbmax = 2000 e uso dos movimentos V1 e V2 com intensificação.

**TABELA 22** - Melhores resultados para o problema 1.

Problema 1						
	Média		Desvio Padrão		Melhor	
	Sem Uso Intens.	Com Uso Intens.	Sem Uso Intens.	Com Uso Intens.	Sem Uso Intens.	Com Uso Intens.
V1	657,55	650,95	24,79	28,30	590,76	580,55
V2	582,08	570,30	21,16	22,21	544,75	542,31
V1,V2	537,36	542,01	8,56	13,62	526,76	525,42

**TABELA 23** - Melhores resultados para o problema 2.

Problema 2						
	Média		Desvio Padrão		Melhor	
	Sem Uso Intens.	Com Uso Intens.	Sem Uso Intens.	Com Uso Intens.	Sem Uso Intens.	Com Uso Intens.
V1	951,07	943,02	21,68	25,84	915,51	905,61
V2	895,75	883,03	20,67	21,71	870,18	852,13
V1,V2	867,96	863,06	13,00	14,70	852,39	847,82

**TABELA 24** - Melhores resultados para o problema 3.

Problema 3						
	Média		Desvio Padrão		Melhor	
	Sem Uso Intens.	Com Uso Intens.	Sem Uso Intens.	Com Uso Intens.	Sem Uso Intens.	Com Uso Intens.
V1	954,01	948,12	19,60	16,95	912,15	911,63
V2	903,63	901,17	14,50	17,84	871,29	868,49
V1,V2	879,90	870,10	15,05	20,73	846,39	837,79

**TABELA 25** - Melhores resultados para o problema 4.

Problema 4						
	Média		Desvio Padrão		Melhor	
	Sem Uso Intens.	Com Uso Intens.	Sem Uso Intens.	Com Uso Intens.	Sem Uso Intens.	Com Uso Intens.
V1	1215,35	1210,79	12,79	10,70	1188,14	1183,26
V2	1124,93	1118,83	15,66	12,52	1100,37	1098,28
V1,V2	1087,72	1079,54	18,15	16,70	1062,39	1061,07



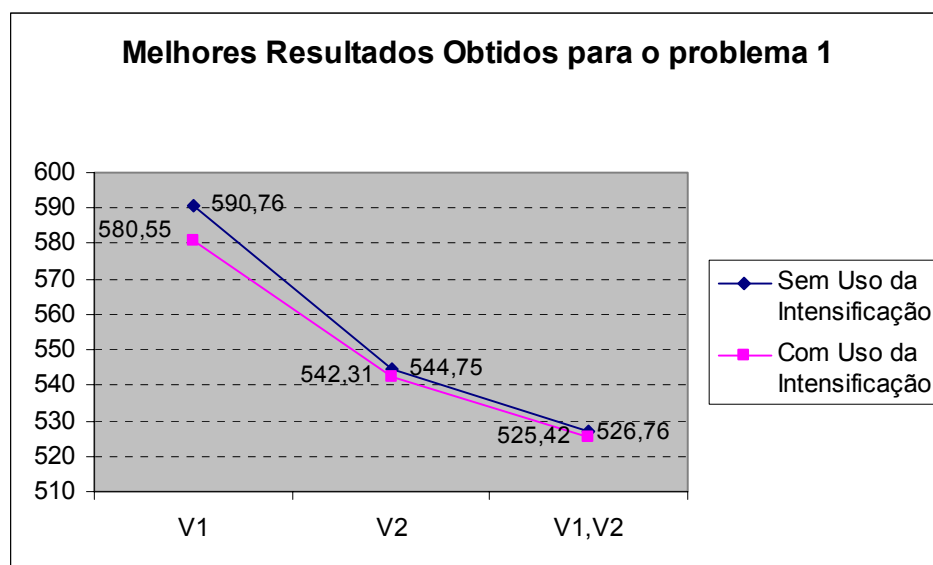
**TABELA 26** - Melhores resultados para o problema 5.

Problema 5						
	Média		Desvio Padrão		Melhor	
	Sem Uso Intens.	Com Uso Intens.	Sem Uso Intens.	Com Uso Intens.	Sem Uso Intens.	Com Uso Intens.
V1	1569,11	1561,59	10,12	16,22	1547,53	1521,34
V2	1421,41	1393,56	34,40	12,75	1360,37	1356,3
V1,V2	1387,93	1377,82	26,64	16,37	1355,34	1352,74

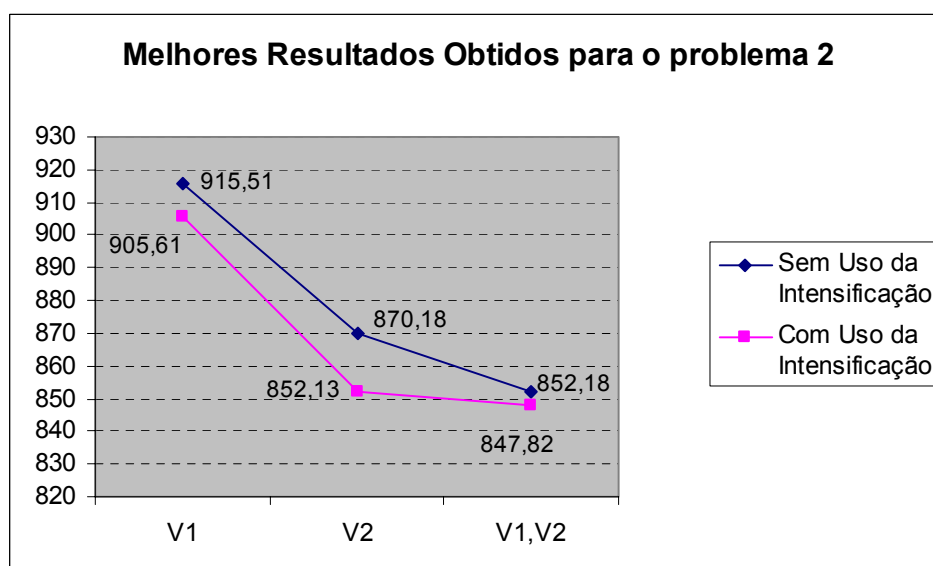
Para o problema 1, houve uma redução do resultado médio dos movimentos agrupados sem intensificação de 18,27% para V1 e 7,68% para V2. Com intensificação esta redução foi de 16,73% para V1 e 4,96% para V2. Para o problema 2 a redução do resultado médio foi de 8,73% para V1 e 3,10 % para V2. Com intensificação esta redução foi de 8,47% para V1 e 2,26 % para V2. Para o problema 3 a redução do resultado médio foi de 7,76% para V1 e 2,62% para V2. Com intensificação esta redução foi de 8,22% para V1 e 3,44% para V2. Para o problema 4 a redução do resultado médio foi de 10,58% para V1 e 3,30% para V2. Com intensificação esta redução foi de 10,32% para V1 e 3,51% para V2. Para o problema 5 houve uma redução de 11,54% dos movimentos agrupados sem intensificação para V1 e 2,35% para V2. Com intensificação esta redução foi de 11,08% para V1 e 0,26% para V2. Estes valores percentuais mostram a qualidade superior dos resultados gerados pelos movimentos agrupados (pois o uso de V1 e V2 juntos reduziu a média de todos os conjuntos de experimentos) e destacam a qualidade do movimento V2 em relação ao movimento V1 uma vez que o percentual de melhora do V1 e V2 juntos para V1 foi maior do que o percentual de melhora em relação a V2 para todos os casos.

Com relação aos desvios-padrão dos resultados, nota-se que todos os desvios-padrão são inferiores a 10% do valor das médias, indicando que houve pouca variação dos resultados

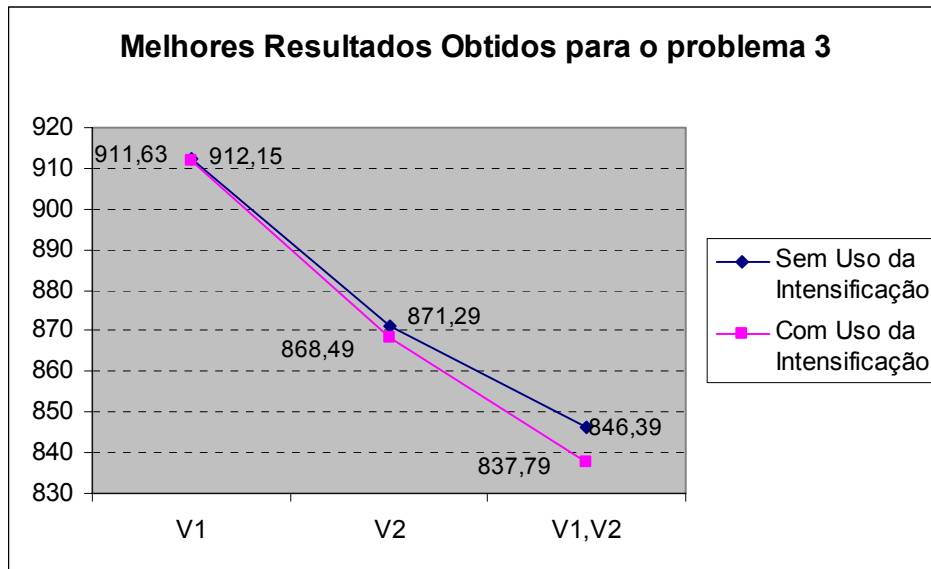
gerados com relação à média. As figuras 54 a 58 ilustram os valores dos melhores resultados obtidos com e sem o uso da estratégia de intensificação.



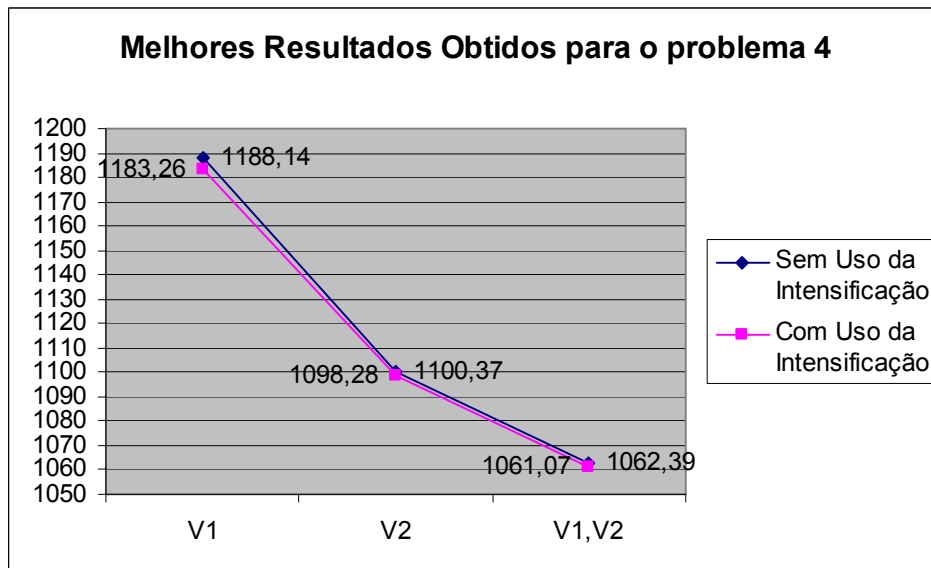
**FIGURA 54** – Melhores resultados obtidos para o problema 1.



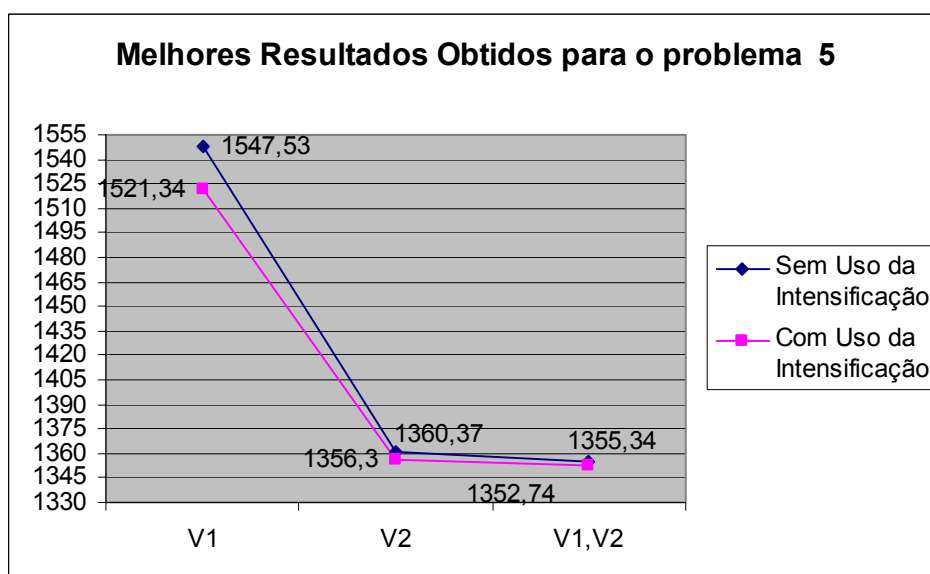
**FIGURA 55**– Melhores resultados obtidos para o problema 2.



**FIGURA 56** – Melhores resultados obtidos para o problema 2.



**FIGURA 57** – Melhores resultados obtidos para o problema 4.



**FIGURA 58** – Melhores resultados obtidos para o problema 5.

Conforme visualizado nas tabelas 22 a 26, as figuras ilustram a melhora que a estratégia de intensificação proporciona ao processo de busca, permitindo encontrar para todos os problemas testados soluções melhores, independente do movimento utilizado.

### 6.3 VALIDAÇÃO

Sendo o PRV um problema *NP-Hard*, geralmente a etapa de validação consiste em comparar os resultados gerados pelo modelo com os resultados de outros trabalhos a fim de mensurar a diferença entre as soluções. Uma vez que limites inferiores e valores ótimos geralmente não estão disponíveis para as instâncias do PRV, grande parte dos trabalhos apresenta como método de validação a comparação com os melhores resultados conhecidos (Cordeau *et al*, 2002). Neste trabalho, o método utilizado para validar o modelo consiste em comparar os resultados obtidos com os resultados publicados por outros autores. Os resultados deste trabalho utilizados na comparação foram obtidos com o uso de  $N_{bmax} = 2000$ , Lista Tabu = 200 e com

uso dos movimentos V1 e V2 com intensificação. Foram selecionados heurísticas clássicas e trabalhos que utilizam a Busca Tabu. Os seguintes trabalhos foram selecionados:

WL: Willard (1989);

PF: Pureza e França (1991);

OM1: Osman (1991);

OM2: Osman (1993);

RG: Rego (1998);

GHL: Gendreau, Hertz e Laporte (1994);

BO: Barbarasolgu e Ozgur (1999);

XK: Xu e Kelly (1996);

TV: Toth e Vigo (2003);

CW: Clarke e Wright (1964);

GM: Gillet e Miller (1974);

MJ: Mole e Jamenson (1976);

CMT: Christofides, Mingozzi e Toth (1979);

T: Taillard (1993);

Foram realizadas comparações envolvendo o resultado da solução e o tempo de processamento. Os resultados das comparações são expostos nas tabelas 28 a 31 e nas figuras 62 a 66. As tabelas 27 e 28 apresentam o resultado das soluções dos trabalhos que utilizaram a

Busca Tabu. Os trabalhos são apresentados indicados na primeira coluna, nas colunas seguintes são mostrados o valor do melhor resultado e a diferença percentual entre o resultado obtido nesta aplicação com relação ao resultado do trabalho indicado (gap). O sinal (+) depois do resultado indica que o resultado deste trabalho é maior que o resultado do trabalho comparado. O sinal (-) depois do resultado indica que o resultado deste trabalho é menor que o resultado do trabalho usado na comparação. Os resultados apresentados a seguir foram extraídos de Barbarasoglu e Ozgur (1999) e Gendreau *et al* (1994).

**TABELA 27** – Comparação dos resultados para os problemas 1,2 e 3 com trabalhos de Busca Tabu.

	Problema 1		Problema 2		Problema 3	
	Best	% Gap	Best	% Gap	Best	% Gap
WL	588	11,91 (-)	893	5,33 (-)	906	8,14 (-)
RG	557,86	6,17 (-)	847	0,10 (+)	832,04	0,69 (+)
PF	536	2,01 (-)	842	0,69 (+)	851	1,58 (-)
OM1	524,61	0,15 (+)	844	0,45 (+)	835	0,33 (+)
OM2	524,61	0,15 (+)	844	0,45 (+)	838	0,03 (-)
GHL	524,61	0,15 (+)	835,77	1,42 (+)	829,45	1,00 (+)
BO	524,61	0,15 (+)	836,71	1,31 (+)	828,72	1,08 (+)
XK	524,61	0,15 (+)	835,26	1,48 (+)	826,14	1,39 (+)
TV	524,61	0,15 (+)	838,60	1,09 (+)	828,56	1,10 (+)

**TABELA 28** – Comparação dos resultados para os problemas 4 e 5 com trabalhos de Busca Tabu.

	Problema 4		Problema 5	
	Best	% Gap	Best	% Gap
WL	-	-	-	-
RG	1047,21	1,31(+)	1352,88	0,014(-)
PF	1081	1,88(-)	-	-
OM1	1052	0,85(+)	1354	0,09(-)
OM2	1044,35	1,58(+)	1334,55	1,34(+)
GHL	1036,16	2,35(+)	1322,65	2,22(+)
BO	1043,89	1,62(+)	1306,16	3,44(+)
XK	1029,56	2,97(+)	1298,58	4,00(+)
TV	1028,42	3,08(+)	1291,45	4,53(+)

Para realizar as comparações com as heurísticas clássicas, foram selecionados os trabalhos apresentados nas tabelas 29 e 30. A estrutura utilizada para apresentação dos resultados nas tabelas é idêntica às tabelas anteriores.

**TABELA 29** – Comparação dos resultados para os problemas 1,2 e 3 com heurísticas clássicas.

	Problema 1		Problema 2		Problema 3	
	Best	% Gap	Best	% Gap	Best	% Gap
CW	578,56	10,11 (-)	888,04	4,74 (-)	878,70	4,88 (-)
GM	546	3,92 (-)	865	2,03 (-)	862	2,89 (-)
MJ	575	9,44 (-)	910	7,33 (-)	882	5,28 (-)
CMT	534	1,63 (-)	871	2,73 (-)	851	1,58 (-)

**TABELA 30** – Comparação dos resultados para os problemas 4 e 5 com heurísticas clássicas.

	Problema 4		Problema 5	
	Best	% Gap	Best	% Gap
CW	1204	13,47 (-)	1540	13,84 (-)
GM	1079	1,69 (-)	1389	2,68 (-)
MJ	1259	18,65(-)	1545	14,21(-)
CMT	1093	3,01(-)	1418	4,82(-)

A seguir, as figuras 59 a 63 ilustram a comparação dos resultados obtidos com os resultados dos trabalhos com Busca Tabu e com heurísticas clássicas..





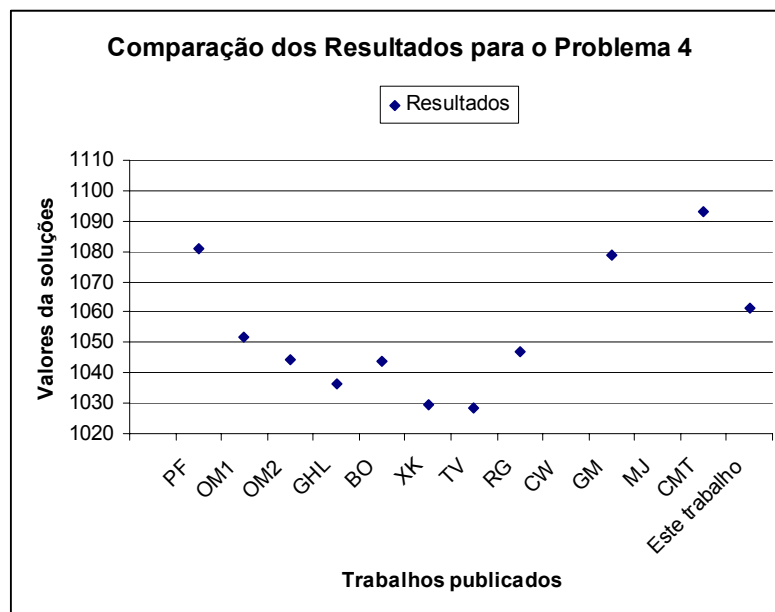
**FIGURA 59** – Comparação dos resultados para o problema 1.



**FIGURA 60** – Comparação dos resultados para o problema 2.



**FIGURA 61** – Comparação dos resultados para o problema 3.



**FIGURA 62** – Comparação dos resultados para o problema 4.



**FIGURA 63** – Comparação dos resultados para o problema 5.

Análise dos dados comparativos mostra que o algoritmo desenvolvido para este trabalho apresenta uma significativa superioridade com relação às heurísticas clássicas, pois conseguiu superar todos os resultados das heurísticas para os problemas utilizados. Com relação aos trabalhos utilizando Busca Tabu este algoritmo mostra-se competitivo com os outros trabalhos uma vez que consegue superar uma média de dois trabalhos por problema. Além disso, os resultados gerados pelo algoritmo desta aplicação excedem em menos de 5% os valores dos outros algoritmos em todos os casos. Dos 35 casos em que o algoritmo proposto excede os outros resultados, em 25 casos esse percentual excedente é menor que 2%. Os experimentos mostram que a qualidade deste algoritmo diminui conforme aumenta o tamanho da instância. Os resultados gerados foram mais homogêneos com os resultados de outros trabalhos utilizados na comparação para as instâncias de até 100 vértices.

## 7 CONSIDERAÇÕES FINAIS

Este trabalho foi desenvolvido com o objetivo de estudar o Problema de Roteamento de Veículos. Basicamente este problema visa criar rotas de custo mínimo para atender um conjunto de clientes. Devido à grande quantidade de características e restrições que podem ser incorporadas ao problema, na literatura é possível encontrar diversas variantes do problema. Neste trabalho foi estudada a versão clássica em que apenas a restrição quanto capacidade do veículo é adotada. Em função de ser um problema *NP-Hard*, grande parte das publicações trabalha com heurísticas e metaheurísticas para resolvê-lo. Neste trabalho optou-se por utilizar a Busca Tabu em função do prestígio desta metaheurística na resolução do PRV.

Foi construída uma aplicação utilizando a Busca Tabu dividida em 4 módulos: i) módulo de geração das redes ii) módulo de geração da solução inicial iii) módulo da Busca Tabu e iv) módulo da validação. O módulo da geração das redes gera a matriz de distância utilizada pela aplicação. O módulo de geração da solução inicial gera uma solução baseada na heurística do vizinho mais próximo para ser utilizada no módulo da Busca Tabu, que executa o algoritmo da Busca Tabu desenvolvido para esta aplicação e no módulo da validação é feita a comparação dos resultados obtidos com os resultados de outros trabalhos publicados. Visando aumentar a flexibilidade da Busca Tabu, foram definidos dois tipos de movimentos geradores de vizinhança: V1 que se baseia na troca de vértices entre rotas e V2 que se baseia na realocação de vértices. Estes movimentos ganham denominação de V1' e V2' quando utilizados na estratégia de intensificação, pois ao serem executados aumentam o tamanho da vizinhança gerada. Visando diversificar as soluções geradas pelos movimentos optou-se por executar os movimentos isolados e agrupados. Para tornar o processo de busca mais robusto foi adotada uma estratégia de intensificação que visava aumentar o tamanho da vizinhança gerada ao redor das boas soluções já encontradas durante o processo de busca. Esta estratégia foi implementada através do auxílio de uma Lista de Soluções Elite que mantinha a relação das melhores soluções encontradas. Para

realizar os experimentos foram selecionados problemas clássicos da literatura amplamente utilizados para testes.

O algoritmo desenvolvido neste trabalho foi construído com o intuito de permitir a fácil utilização e combinação dos movimentos geradores de vizinhança, bem como facilitar a incorporação de novos procedimentos. Por esta razão o algoritmo foi construído em módulos que interagem entre si e podem ser facilmente modificados. Essa característica de desenvolvimento tornou o algoritmo mais flexível e robusto. Facilmente é possível integrar outros mecanismos geradores de vizinhança, bem como outras rotinas ao código já existente. Buscando melhorar os resultados obtidos, outros módulos poderiam ser agregados ao algoritmo existente, como por exemplo, um módulo para aplicação de rotinas de otimização local nas rotas geradas. Baseado na literatura (Gendreau, Hertz e Laporte, 1994; Xu e Kelly,1996; Taillard,1993), acredita-se que melhorias significativas podem ser obtidas com o uso de rotinas de otimização local aplicadas ao rearranjo dos clientes das rotas geradas. Nesse caso heurísticas de otimização aplicadas ao PCV (Lin e Kernighan,1971; Laporte, 1992b) poderiam ser utilizadas nessa etapa. Outra abordagem interessante na busca de melhores resultados seria implementar a diversificação de soluções iniciais, permitindo assim o uso de diferentes soluções geradas a partir de outras técnicas além da heurística do vizinho mais próximo (Tyagi,1968) que foi utilizada.

Diversos experimentos foram realizados através dos quais foi possível realizar análises com relação aos melhores resultados gerados por valor de Nbmax, por tamanho da Lista Tabu e por movimento utilizado. Também foi possível comparar a qualidade do processo de busca com e sem o uso da estratégia de intensificação. Os experimentos realizados mostram que a qualidade da política V2, que se baseia na realocação de vértices é superior a política V1, que realiza a troca de vértices. Conforme a discussão apresentada, V1 apresenta mais restrições para a geração de movimentos, o que impede que ele gera uma vizinhança grande e diversificada. O movimento V2 apresenta um número menor de restrições e em função disso permite a geração de uma vizinhança maior. Vizinhança maior implica em maior espaço de busca percorrido e desta

maneira, maior possibilidade de encontrar melhores resultados. Da mesma maneira, os experimentos mostram que a qualidade dos resultados gerados pelos movimentos agrupados é superior aos gerados pelos movimentos isolados. V1 e V2 juntos geram uma diversidade muito maior de vizinhos em função das diferentes dinâmicas que os movimentos apresentam e que permitem alterar soluções geradas por um movimento com o uso do outro movimento, criando soluções fora do padrão esperado pelo movimento. Nota-se através dos experimentos que a qualidade do processo de busca é superior quando é se usa o processo de busca com o uso da estratégia de intensificação. Também é possível observar que a estratégia de intensificação gera melhores resultados quando aplicada à movimento flexíveis que permitem a execução do aumento da vizinhança que a estratégia proporciona.

Geralmente a validação de problemas *NP-Hard* é realizada através da comparação dos resultados obtidos com os resultados publicados de outros autores. Para que fosse possível realizar esta comparação foram selecionados alguns problemas do clássico conjunto de 14 problemas de Christofides, Mingozzi e Toth (1979). Para validar o modelo desenvolvido, foram selecionadas heurísticas clássicas de resolução do PRV e trabalhos que utilizaram a Busca Tabu como ferramenta para solucionar o problema. Atendendo as expectativas iniciais a qualidade dos resultados gerados pelo algoritmo proposto mostrou-se superior às heurísticas clássicas, reafirmando a superioridade das metaheurísticas, que incorporam estruturas de memória e outros recursos, em gerar soluções de maior qualidade. Com relação aos trabalhos que utilizaram a Busca Tabu, os resultados gerados pelo algoritmo proposto não conseguiram superar os resultados apresentados pelos outros trabalhos utilizados na validação, apresentando a menor diferença entre os resultados para o problema de 1, com a 50 vértices. Ao buscar uma justificativa para a qualidade das soluções geradas, comparando este algoritmo com os outros trabalhos, nota-se que os demais trabalhos utilizam uma série de recursos adicionais para otimização que não foram implementados neste algoritmo. Estruturas sofisticadas de geração de vizinhança (Gendreau, Hertz e Laporte, 1994; Rego e Roucairol,1996; Rego,1998), procedimentos para pós-otimização das rotas (Gendreau, Hertz e Laporte, 1994; Xu e Kelly,1996; Taillard,1993 ), regras

para eliminação de grandes distâncias entre os vértices (Toth e Vigo,2003), auxiliam na obtenção de uma busca mais qualificada, com a geração de melhores resultados. Desta maneira pode-se afirmar que o algoritmo proposto é bastante competitivo para as instâncias pequenas, diminuindo a qualidade em comparação aos outros trabalhos com Busca Tabu para instâncias maiores.

A maior contribuição deste trabalho se refere á constatação da importância da utilização de movimentos flexíveis e não engessados no processo de busca e a visualização dos benefícios que uma estratégia de intensificação proporciona a um processo de busca.

Como trabalhos futuros, diversas oportunidades são vislumbradas. Implementar regras que tornem a geração de vizinhança mais robusta e sofisticada pode contribuir para o aumento da qualidade da busca, através da geração de vizinhanças grandes e diversificadas. Adotar uma estratégia de diversificação para buscar soluções em regiões desconhecidas do espaço de busca também é uma idéia interessante uma vez que os experimentos mostraram que uma vizinhança mais diversificada gera resultados de maior qualidade. Com relação às características do problema estudado, percebe-se a importância de se trabalhar com um número maior de restrições, como o tempo de duração das rotas ou janelas de tempo, por exemplo, ou adotar outras características, como frota de tamanho heterogêneo ou múltiplos depósitos com o intuito de tornar o problema mais aplicável à realidade.

Sendo o PRV um problema clássico de Otimização Combinatória, a bibliografia sobre o assunto é extensa e diversificada. O fato do PRV permitir a abordagem de diferentes características (um depósito, múltiplos depósitos, frota homogênea, frota heterogênea) e diferentes restrições (capacidade do veículo, tempo de duração da rota, janelas de tempo) permite o estudo dirigido do problema focado na modelagem utilizada e o desenvolvimento de algoritmos específicos para atender as características/restrições adotadas. Entretanto, devido à semelhança dos diversos tipos de problemas, geralmente, pequenas adaptações podem ser feitas para adequar os algoritmos a um tipo específico do problema. Este fato proporciona aos estudiosos do PRV uma fonte rica e diversificada para abordagem deste problema.

## 8 REFERÊNCIAS BIBLIOGRÁFICAS

AARTS, E.H.L., LENSTRA, J.K. **Local Search in Combinatorial Optimization**. Wiley, Chichester, 1997.

ALI A.I., KENNINGTON J.L. The asymmetric m-traveling salesman problem: a duality based branch-and-bound algorithm. **Discrete Applied Mathematics** 13:259-276, 1986.

ALFA, A.S., HERAGU, S.S., CHEN, M. A 3-opt based simulated annealing algorithm for vehicle routing problems. **Computers & Industrial Engineering**, v.21, p.635-639, 1991.

BAKER, B. M., AYECHIEW, M.A. A genetic algorithm for the vehicle routing problem. *Computers and Operations research*, v.30, p.787-800, 2003,

BALINSK, M.L., QUANDT, R.E. On an integer program for a delivery problem, *Operations Research*, v.12, p.300-304, 1964

BALLOU, R. H. **Logística Empresarial – Transportes, Administração de Materiais e Distribuição Física**, São Paulo: Atlas, 1993,

BALLOU, R. H. **Gerenciamento da Cadeia de Suprimentos – Planejamento, Organização e Logística Empresarial**, 4Ed, Porto Alegre: Bookman, 2001,

BARBAROSOGLU, G.; OZGUR, D. A tabu search algorithm for the vehicle routing problem. **Computers & Operations Research** v.26, p.255-270, 1999.

BARTHOLDI III, J.J., PLATZMAN, L.K., LU, C.R., WARDEN, W.W. A minimal technology routing system for meals on wheels. **Interfaces** v.13, p.1-8, 1983.

BEKTAS, T. The multiple traveling salesman problem: an overview of formulations and solution procedures. *Omega*, v.34, p.209-219, 2006.



BELLMANN, R.E. On a Routing Problem. **Quarterly Applied Mathematics**, v.16, p.87-90, 1958.

BELLMORE, M., NEMHAUSER, G.L. The Traveling Salesman Problem: A survey. **Operations Research**, v.16, p.538-558, 1968.

BRASSARD, G. BRATLEY, P. **Fundamentals of Algorithmics**. EUA: Prentice Hall, 1996.

BREEDAM, A. V. Comparing descent heuristics and metaheuristics for the vehicle routing problem. **Computer & Operations Research** v.28, p.289-215, 2001.

BREEDAM, A. V. Improvement heuristics for the Vehicle routing Problem based on Simulated Annealing. **European Journal of Operational Research**, v.86, p.480-490, 1995.

BODIN, L.D., GOLDEN, B.L., ASSAD, A.A., BALL, M.O. Routing and Scheduling of vehicles and crews: The State of the Art. **Computers and Operations Research** v.10, p.69-211, 1983.

CLARKE, G; WRIGHT, J.W. Scheduling of vehicles from a central depot to a number of delivery points. **Operations Research** v.12, p.568-581, 1964.

CEDER, A., STERNY, H.I. Deficit Function Bus Scheduling with Deadheading trip insertion for fleet size reduction. **Transportation Science** v.15, p.338-363, 1981.

CORDEAU, J-F., Gendreau, M., LAPORTE, G., POTVIN, J.-Y., SEMET, F. A guide to vehicle routing heuristics. **Journal of the Operational Research Society**, v.53, p.512-522, 2002.

COOK, W. J; CUNNINGHAM, W. H; PULLEYBLANK, WILLIAM R; SCHRIJVER, A.. **Combinatorial Optimization**. Willey, 1998.

CHRISTOFIDES, N., MINGOZZI, A., TOTH P. Exact algorithms for the vehicle routing problem, based on spanning tree and shortest path relaxations. **Mathematical Programming** v.20, p.255-282, 1981.

CHRISTOFIDES, N., MINGOZZI, A., TOTH P. The Vehicle Routing Problem. In: Christofides, Nicos. Combinatorial Optimization UMI, 1979

CHRISTOFIDES, N., EILON, S. An Algorithm for the vehicle Dispatching Problem. **Operational Research Quartely**, v. 20, n.3, 1969.

CHRISTOFIDES, N. The Traveling Salesman problem. In: Christofides, N., Mingozi, A., Toth, P., Sandi, C. Combinatorial optimization. Londres: John Wiley and Sons, 1979

CZECH, Z.J., Solving the set-partitioning problem using parallel simulated annealing, The 3rd International Conference on Parallel Processing and Applied Mathematics (PPAM'99), Kazimierz Dolny, Poland, (September 14-17, 1999), 256-265.

DANTZIG, G.B., RAMSER, J.H. The truck dispatching problem. **Management Science**, v.6, p.80-91, 1959.

DANTZIG, G.B., FULKERSON D.R., JOHNSON S.M. Solution of a large-scale traveling salesman problem. **Operations Research**, v.2, p.393-410, 1954.

DUECK,G. New optimization heuristics: The great deluge algorithm and the record-top-record travel. **Journal of Computational Physics**, v.104, p.86-92, 1993.

DUECK,G., SCHEURER,T. Threshold accepting: A general purpose optimization algorithm. **Journal of Computational Physics**, v.90, p.161-175, 1990.

EILON,S., WATSON-GANDY,C.D.T., CHRISTOFIDES,N. Distribution Management: Mathematical Modelling and Pratical Analysis. Griffin, Londres, 1971.

FISHER, M.L. Optimal solution of vehicle routing problems using minimum k-trees. **Operations Research** v.42, n.4, p.626-642, 1994.

FISHER, M.L., JAIKUMAR, R. A generalized assignment heuristic for vehicle routing. **Networks** v.11, p.109-124, 1981.

FOGEL D.B. A parallel processing approach to a multiple traveling salesman problem using evolutionary programming. In: Proceeding of the fourth annual symposium on parallel processing. Fullerton, CA, 1990 p.318-326.

GAVISH B., SRIKANTH K. An optimal solution method for large-scale multiple traveling salesman problem. **Operations Research** v.34, n.5, p.698-717, 1986.

GARFINKEL, R.S. Branch and bound Methods for Integer Programming. In: Christofides, Nicos. Combinatorial Optimization UMI, 1979

GENDREAU, Michel, POTVIN, Jean-Yves. Metaheuristics in Combinatorial optimization. *Annals of Operations Research*, v.140, p.189-213, 2005.

GENDREAU, M., HERTZ, A., LAPORTE, G.. A Tabu Search Heuristic for the Vehicle Routing Problem. **Management Science** v.40, p.1276-1290, 1994.

GENDREAU, M., LAPORTE, G. POTVIN, J-Y. Metaheuristics for the vehicle routing problem. Technical Report, Centre de Recherche sur les transports, Université de montreal, 1999.

GILLET, B.E; MILLER, L.R. A heuristic algorithm for the vehicle dispatch problem. **Operations Research** v.22, p.240-349, 1974.

GORENSTEIN S. Printing press scheduling for multi-edition periodicals. **Management Science** v.16, n.6, p.373-383, 1970.

GOLDEN, B.L; MAGNANTI, T.L.; NGUYEN, H.Q. Implementing Vehicle Routing Algorithms. **Networks** v.7, p.113-148, 1977.

GOLDEN,B.L, WASIL, E.A.KELLY JP, CHAO I-M. The impact of metaheuristics on solving the vehicle routing problem: algoritms, problem sets, and computational results. In: Crainic T, Laporte G, editors. Fleet management and logistic. Boston, MA: Kluwer, 1998, p.33-56.

GOLDBARG, M.C.,LUNA, H.P. Otimização Combinatória e Programação Linear: Modelos e Algoritmos. Campus, 2000.

GLOVER, FRED. Future paths for integer programming and links to artificial intelligence. **Computers and Operations Research** v.1, p.533-549, 1986.

GLOVER, FRED. Tabu Search – parte 1. *ORSA Journal on Computing* v.1, n.3, 1989.

GLOVER, FRED. Tabu Search – parte 2. *ORSA Journal on Computing* v.2, n.1, 1990.

GLOVER,Fred; LAGUNA, Manuel. Tabu Search. Kluwer Academic Publishers, 1997.

HADJICONSTANTINO, E., CHRISTOFIDES, MINGOZZI, A. A new Exact algorithm for the vehicle routing problem based on q-Path and k-Shortest Path Relaxation. **Annals of Operations Research** v.61, p.21-44, 1995

HILLIER, Frederick S.; LIEBERMAN, Gerald J. Introduction to Operations Research, Campus, 1988.

HO, S.C, HAUGLAND, D. A tabu search heuristic for the vehicle routing problem with time windows and split deliveries. **Computers & Operations Research** v.31, p.1947-1964, 2004

HOLLAND, J.H. Adaptation in natural artificial systems. University of Michigan Press, 1975.

HSU C., TSAI M., CHEN W. A study of feature-mapped approach to the multiple traveling salesman problem. *IEEE International Symposium on Circuits and Systems* v.3, p.1589-1592, 1991.

KAWAMURA, H., YAMAMOTO,M.,MITAMURA,T., SUZUKI,K., OHUCHI,A. Cooperative search on pheromone communication for vehicle routing problems. *IEEE Transactions on Fundamental*, E81-A.1089-1096, 1998.

KYTOJOKI, J, NUORTIO, T. BRAYSY, O., GENDREAU, M. An efficient variable neighborhood search heuristic for very large scale vehicle routing problems. **Computers & Operations Research**, 2005.

KIRKPATRICK,S., GELLATT, C.D. Jr.,VECCHI, M.P. Optimization by simulated annealing. *Science*, v.20, n 4598, p.671-680, 1983.

KULCAR,T, Optimizing Solid Waste Colletion in Brussels, **European Journal of Operational Research**, v.90, p.71-77, 1996

LAPORTE,G., GENDREAU,M., POTVIN,J., SEMET, F. Classical and modern heuristics for the vehicle routing problem. **Intl.Trans. in Op. Res** v.7, p.285-300, 2000.

LAPORTE, G. The Vehicle Routing Problem: An overview of exact and approximate algorithms. **European Journal of operational Research** v.59, p.345-458, 1992a.

LAPORTE, G. The Traveling Salesman Problem: An overview of exact and approximate algorithms. **European Journal of operational Research** v.59, p.231-247, 1992b.

LAPORTE, G. SEMET, F. Classical Heuristics for the Vehicle Routing Problem. *Les Cahiers du GERAD*, G98-54, Group for Research in Decision Analysis, Montreal, Canada, 1998.

LAPORTE, G., NOBERT, Y. A cutting planes algorithm for the m-salesman problem. **Journal of the Operational Research Society** v.31, p.1017-1023,1980.

LAPORTE, G., MERCURE, H., NOBERT, Y. A branch-and-bound algorithm for a class of asymmetrical vehicle routing problem. **Journal of the Operational Research Society** v.31, p.1017-1023,1986.

LAPORTE, G., NOBERT, Y. Exact algorithms for the vehicle routing problem. In: S. Martello, G. Laporte, M. Minoux and C. Ribeiro (eds). *Surveys in Combinatorial Optimization*, North-Holland, Amsterdam, 147-184, 1987.

LAWER, E.L., LENSTRA, J.K., RINNOOY KANN, A.H.G., SHMOYS, D.B. *The Traveling Salesman Problem*. Chichester: Wiley; 1985.

LENSTRA, J.K., RINNOOY Kan, G. Complexity of Vehicle Routing and Scheduling Problems. **Networks** v.11, p.221-227, 1981.

LI, F., BRUCE, G., WASIL, E. Very large scale vehicle routing: new test problems, algorithms, and results. **Computers & Operations Research** v.32, p.1165-1179, 2005.

LIN, S., KERNIGHAM, B.W. An effective implementation for the traveling salesman problem. **Operations Research**, v.21, p.498-516, 1973.

LIN, S. Computer solution of the traveling salesman problem. *Bell System Technical Journal* 44, 2245-2269, 1965.

LI, F., GOLDEN, B., WASIL, E. Very large-scale vehicle routing: new test problems, algorithms and results. **Computers & Operations Research**, v.32, p.1165-1179, 2005.

MAMBER, U. *Introduction to Algorithms - A Creative Approach*. Wesley, 1989.

MALMBORG, C.J., SIMONS, G.R. Integrating logistical and Processing Functions through mathematical modeling. **Applied Mathematical Modelling** v.13, n6, p.357-364, 1989.

MARINAKIS, Y., MIGDALAS, A. Heuristic Solutions of Vehicle Routing Problems in Supply Chain Management. *Combinatorial and Global Optimization*, P.M. Pardalos et al. (ed.s), 2002, Scientific World, p. 205-235.

MARTINHON, C., LUCENA, A., MACULAN, N. Stronger K-tree relaxations for the vehicle routing problem. **European Journal of Operational Research** v.158, p.56-71, 2004.

MATSUYAMA, Y. Self-organization via competition, cooperation and categorization applied to extended vehicle routing problem. In: Proceeding of International Joint Conference on Neural Networks, p I-385-390, Seattle, WA, 1991.

MESTER, D., BRAYSY, O. Active-guided evolution strategies for large-scale capacitated vehicle routing problems. **Computers & Operations Research** v.32, p.1593-1614, 2005.

METROPOLIS, N.A.W.,M.N., Rosenbluth, *et al.* Equation of state calculations by fast computing machine. **Journal Chemical and Physical**, v.21, p.1087-1092, 1953.

MOLE, R.H; JAMENSON, S.R.A sequential route-building algorithm employing a generalized savings criterion. **Operations Research Quarterly**. V.27, p.503-511, 1976.

NEWTON, R, THOMAS,W. Bus Routing in Multischoll System, **Computers & Operational Research**, v.1, p.213-222, 1974.

OSMAN, I.H., LAPORTE,G. Metaheuristics: A bibliography. *Annals of Operations Research* 63, 513-628, 1996.

OSMAN, I.H. Metastrategy simulated annealing and tabu search algorithms for combinatorial optimization problems. **Annals of Operations Research**, 41:421-451, 1993

OSMAN, I.H. Simulated annealing and tabu search algorithms for combinatorial optimization problems. Ph.D.Dissertation, The management School, Imperial College, London,1991

PAPADIMITRIOU, C.,STEIGLITZ, K. Combinatorial optimization: algorithms and complexity. Prentice Hall, 1982.

PERLS, J., DASKIN, M.S. A warehouse location-routing problem. **Transportation Research**, v.19B, p.381-396, 1985.

PUREZA, V.M., FRANÇA, P.M. Vehicle routing problems via tabu search metaheuristic, Publication CRT-747, Centre de recherche sur les transports, Montreal, 1991.

PHAM, D.T., KARABOGA, D. Intelligent Optimisation Techniques. Springer, 2000.

PRINS, C. A simple and effective evolutionary algorithm for the vehicle routing problem. **Computers & Operations Research** v.31, p.1985-2002, 2004.

RAYWARD-SMITH V, OSMAN I., REEVES C., SMITH, G. Modern heuristic search methods. Chichester: Wiley; 1996.

RALPH, T.K., KOPMAN, L., PULLEYBLANK, W.R., TROTTER, L.E. On the Capacitated Vehicle Routing Problem. **Mathematical programming**, v.94, p.343-359, 2001.

REEVES, C. R. Modern Heuristic Techniques. In: Rayward-Smith, V.J.; Osman, I.H.; Reeves, C.R.; Smith, G.D. Modern Heuristic Search Methods. Wiley, 1996.

REGO, C. A subpath ejection method for the vehicle routing problem. **Management Science**, v.44, p.1447-1459, 1998.

REGO, C., ROUCAIROL, C. A parallel tabu search algorithm using ejection chains for the vehicle routing problem. In: Osman I.H. and Kelly J.P (eds) *Meta-Heuristics: Theory and Applications*. Kluwer: Boston, 661-675, 1996.

REINELT, G. The travelling salesman problem, Computational solutions for TSP Applications. Springer-Verlag, 1994.

ROCHAT, Y., TAILLARD, E.D. Probabilistic diversification and intensification in local search for vehicle routing. **Journal of Heuristics**, v.1, p.147-167, 1995.

ROBUSTÉ, F., DAGANZO, C.F., SOULEYRETTE II, R. Implementing vehicle models. **Transportation Research**, 24B.263-286, 1990.



RODRIGUES, A. G., GÓMEZ, A. T.; Utilização de metaheurísticas para a modelagem de problemas de seleção de partes e scheduling em um sistema de manufatura flexível. **Scientia**, São Leopoldo, v.11, n.2,p. 73-108, 2000.

RUSSEL R.A. An effective heuristic for the m-tour traveling salesman problem with some side conditions. **Operations Research** v.25, n.3, p.517-524, 1977.

RYAN, J.L., BAILEY T.G., MOORE, J.T., CARLTON, W.B. Reactive Tabu search in unmanned aerial reconnaissance simulations. Proceeding of the 1998 winter simulation conference, v1, 1998, p.873-879.

SKIENA, Steven S. The Algorithm Design Manual. New York: Springer-Verlag, 1997.

SZWARCFITER, Jayme Luiz. Grafos e algoritmos computacionais. Rio de Janeiro: Campus, 1984.

SVESTKA J.A., HUCKFELDT V.E. Computational experience with an m-salesman traveling algorithm. **Management Science** v.19, n.7, p.790-799, 1973.

SONG, C., LEE, K., LEE, W.D. Extended simulated annealing for augmented TSP and multi-salesman TSP. Proceeding of the international joint conference on neural networks, v.3, 2003, p.2340-2343.

TAILLARD, É.D. Parallel iterative search methods for the vehicle routing problems. **Networks** v.23, p.661-673, 1993.

TANG L., LIU J., RONG A. YANG Z. A multiple traveling salesman problem model for hot rolling scheduling in Shangai Baoshan Iron & Steel Complex. **European Journal of Operational Research** v.124, p.267-82, 2000.

TARANTILIS, C.D., IOANNOU, G., PRASTACOS, G. Advanced vehicle routing algorithms for complex operations management problems **Journal of Food Engineerig**, v.70, p.455-471, 2005

TARANTILIS, C.D., KIRANOUDIS CT. BoneRoute: an adaptative memory-based method for effective fleet management. *Annals of Operations Research*, v.115, p.227-241, 2002.

THANGIAH, S.R; PETROVIC, P. Introduction to Genetic Heuristics and vehicle Routing Problems with Complex Constraints. In: Woodruff, David, L. Advances in Computational and Stochastic Optimization, Logic programming , and Heuristic search: Interfaces in Computer Science and Operations research. Kluwer Academic Publishers, 1997.

TIMLIN, M.T., PULLEYBLANK, W.R. Precedence Constrained Routing and Helicopter Scheduling: Heuristic Design. Department of Combinatoric and Optimization University of Waterloo, N2L 3G1, Canadá, 1990.

TOMPHSON, P.M., PSARAFITIS, H.N. Cyclic transfer algorithms for the multivehicle routing and scheduling problems. **Operations Research**, v.41, p.935-946, 1993.

TOSCANI, L. V, VELOSO, P. A.S. Complexidade de Algoritmos. Porto Alegre: Sagra Luzzato, 2001.

TOTH, P., VIGO, D. Exact solution of the vehicle routing problem. In Crainic TG and Laporte G (eds). *Fleet Management and Logistics*. Kluwer: Boston, p.1-31, 1998.

TOTH, P., VIGO, D. Models, relaxations and exact approaches for the capacitated vehicle routing problem. **Discrete Applied Mathematics** v.123, p.487-512, 2002.

TOTH, P., & VIGO, D. The granular tabu search and its application to the vehicle-routing problem. **INFORMS Journal on Computing**, v.15, n.4, p.333-346, 2003.

TYAGI, M., A Practical Method for the Truck Dispatching Problem. **Journal of the Operations Research Society of Japan**, v.10, p.76-92, 1968.

VIANA, V. Meta-heurísticas e Programação Paralela em Otimização Combinatória. Fortaleza: EUFC, 1998.

XU, J., KELLY, J.P. A Network Flow-Based Tabu Search Heuristic for the Vehicle Routing Problem. **Transportation Science** v.30, p.379-393, 1996.

WACHOLDER E., HAN J., MANN R.C. A neural network algorithm for the multiple traveling salesman problem. **Biology in Cybernetics** v.61, p.11-19, 1989.

WILLARD J.A.G. Vehicle routing using r-optimal tabu search, M.Sc Dissertation. The Management School, Imperial College, London,1989.

YU, Z., JINHAI L. GUOCHANG G., RUBO Z., HAIYAN Y. An implementation of evolutionary computation for path planning of cooperative mobile robots. Proceeding of fourth world congress on intelligent control and automation, v3, p.1798-1802, 2002.

ZHANG T., GRUVER W.A., SMITH M.H. Team scheduling by genetic search. Proceeding of the second international conference on intelligent processing and manufacturing of materials, v2, p 839-844, 1999.