

**DIOGO SEIJIY TSUDA**

**MODELO DE ROTEIRIZAÇÃO DE VEÍCULOS EM UMA EMPRESA  
IMPORTADORA DE PRODUTOS JAPONESES**

Trabalho de Formatura apresentado à Escola  
Politécnica da Universidade de São Paulo para  
a obtenção do Diploma de Engenheiro de  
Produção

São Paulo

2007



**DIOGO SEIJIY TSUDA**

**MODELO DE ROTEIRIZAÇÃO DE VEÍCULOS EM UMA EMPRESA  
IMPORTADORA DE PRODUTOS JAPONESES**

Trabalho de Formatura apresentado à Escola  
Politécnica da Universidade de São Paulo para  
a obtenção do Diploma de Engenheiro de  
Produção

Orientadora:

Prof.<sup>a</sup> Dr.<sup>a</sup> Débora Pretti Ronconi

São Paulo

2007

## **FICHA CATALOGRÁFICA**

**Tsuda, Diogo Seijiy**

**Modelo de roteirização de veículos em uma empresa  
importadora de produtos japoneses / D.S. Tsuda. -- São Paulo,  
2007.**

**96 p.**

**Trabalho de Formatura - Escola Politécnica da Universidade  
de São Paulo. Departamento de Engenharia de Produção.**

**1.Pesquisa operacional 2.Transportes (Otimização)  
I.Universidade de São Paulo. Escola Politécnica. Departamento  
de Engenharia de Produção II.t.**

## **AGRADECIMENTOS**

À Profª Débora pelo incentivo, colaboração, paciência e principalmente pela orientação, que nortearam o desenvolvimento deste trabalho e me deram a energia necessária para seguir em frente.

Ao meu pai, sem o qual não seria possível a realização deste trabalho.

À minha mãe, irmão e familiares, pelo amor, carinho e apoio nos momentos difíceis.

Aos meus amigos, pelo suporte e aprendizado.



## **RESUMO**

O presente trabalho aplica técnicas e conceitos do campo da Pesquisa Operacional para a resolução do problema decisório envolvendo a roteirização de veículos para a realização de entregas de mercadorias ao varejo em uma empresa importadora de produtos japoneses. O método de resolução empregado para tanto é uma adaptação do algoritmo de varredura (*sweep algorithm*), apresentado na literatura por Gillett e Miller (1974) na publicação: “*A Heuristic Algorithm for the Vehicle-Dispatch Problem*”. O problema de roteirização considerado neste trabalho é uma instância do problema clássico, denominada roteirização com entregas fracionadas (SDVRP – *Split Delivery Vehicle Routing Problem*), no qual um cliente pode ser atendido por mais de um veículo.

O algoritmo original de varredura foi adaptado para o ambiente *Microsoft Excel*, funcionando em conjunto com o software de otimização “*What's Best!*”. Tal adaptação tem como vantagem uma fácil utilização e implementação, o que é particularmente importante em uma empresa onde o planejamento dos roteiros é realizado de maneira informal. O modelo desenvolvido foi então aplicado a uma série de casos reais de roteirização ocorridos na empresa, com o objetivo de verificar a eficiência e o desempenho do algoritmo em situações práticas.

Os resultados obtidos mostram que através da aplicação de métodos científicos como ferramenta de auxílio à tomada de decisão em problemas de roteirização, é possível obter economias significativas tanto em termos de custos, quanto em termos de distâncias percorridas pelos veículos. Finalmente, a conclusão do trabalho apresenta uma análise crítica do modelo proposto, que levanta alguns pontos que podem nortear o desenvolvimento de estudos futuros.



## ABSTRACT

This report applies techniques and concepts in the Operations Research field to solve a decision problem regarding vehicle routing for the delivery of goods to retailers in a company specialized in the importing of Japanese products. The methodology employed to achieve this goal is an adaptation of the sweep algorithm, introduced in the literature by Gillett and Miller (1974) in the paper “*A Heuristic Algorithm for the Vehicle-Dispatch Problem*”. The routing problem considered here is an instance of the classical problem, named Split Delivery Vehicle Routing Problem (SDVRP), in which a customer can be served by more than one vehicle.

The original sweep algorithm was adapted to the *Microsoft Excel* environment, working in conjunction with the optimization software “*What’s Best!*”. Such an adaptation has the advantage of being relatively user-friendly and easy to be implemented, which is particularly important in a company where route planning is made in quite an informal way. The model was then implemented in a number of real vehicle routing cases in the company, with the aim of verifying the algorithm’s efficiency and performance in practical situations.

The attained results show that by virtue of the adoption of scientific methods as a decision support tool regarding vehicle routing problems, it is possible to reach substantial savings in terms of costs and distance traveled by the vehicles as well. In conclusion, this report shows a critical analysis concerning the presented model, raising some issues that may guide the development of future studies.



## SUMÁRIO

<b>SUMÁRIO.....</b>	<b>I</b>
<b>LISTA DE TABELAS.....</b>	<b>III</b>
<b>LISTA DE FIGURAS.....</b>	<b>V</b>
<b>INTRODUÇÃO .....</b>	<b>1</b>
<b>1. DESCRIÇÃO DA EMPRESA E DEFINIÇÃO DO PROBLEMA.....</b>	<b>5</b>
1.1. DESCRIÇÃO DA EMPRESA.....	5
1.1.1. Descrição das operações da empresa.....	8
1.1.2. Vínculo com a empresa .....	13
1.2. DEFINIÇÃO DO PROBLEMA .....	13
1.2.1. O processo atual de distribuição .....	14
1.3. OBJETIVO GERAL DO TRABALHO .....	16
<b>2. REVISÃO DA LITERATURA .....</b>	<b>19</b>
2.1. PROBLEMAS DE ROTEIRIZAÇÃO DE VEÍCULOS .....	19
2.1.1. Formulação do problema clássico de roteirização de veículos.....	20
2.1.2. Problema de roteirização de veículos com entregas fracionadas (SDVRP).....	21
2.1.3. Complexidade computacional do SDVRP.....	24
2.2. FORMULAÇÃO MATEMÁTICA DO SDVRP.....	25
2.2.1. Parâmetros e formulação do SDVRP .....	25
2.2.2. Detalhamento do modelo .....	27
<b>3. MÉTODO DE RESOLUÇÃO.....</b>	<b>35</b>
3.1. MÉTODO EXATO PARA RESOLUÇÃO DO SDVRP .....	36
3.2. O ALGORITMO DE VARREDURA .....	43
3.2.1. Funcionamento do algoritmo de varredura.....	45
3.2.2. Adaptação do método da varredura ao SDVRP .....	50
3.3. EXEMPLO REDUZIDO .....	53
3.3.1. Dados do problema .....	53
3.3.2. Solução exata do exemplo reduzido .....	55
3.3.3. Resolução através do método de varredura .....	58
<b>4. LEVANTAMENTO E COLETA DE DADOS.....</b>	<b>65</b>
4.1. DADOS DO PROBLEMA .....	65

4.2. OBTENÇÃO DOS DADOS .....	65
4.2.1. Dados sobre os veículos .....	66
4.2.2. Dados sobre os clientes .....	66
4.2.3. Dados sobre a distribuição .....	69
4.3. COMPILAÇÃO DOS DADOS .....	70
<b>5. APLICAÇÃO DO MODELO.....</b>	<b>73</b>
5.1. MODELAGEM DO ALGORITMO .....	73
5.1.1. Requisitos do modelo.....	73
5.1.2. Rotina de resolução do problema de caixeiro viajante.....	77
5.2. VALIDAÇÃO DO MODELO .....	78
<b>6. ANÁLISE DOS RESULTADOS .....</b>	<b>81</b>
6.1. ESTUDO DE CASO .....	81
6.2. ANÁLISE DOS RESULTADOS DAS SIMULAÇÕES DO MODELO .....	82
6.2.1. Qualidade da solução .....	84
6.2.2. Desempenho computacional .....	88
6.2.3. Exatidão da solução.....	91
6.2.4. Conclusão dos resultados obtidos .....	92
6.3. ANÁLISE CRÍTICA.....	93
<b>7. CONCLUSÕES.....</b>	<b>95</b>
<b>REFERÊNCIAS BIBLIOGRÁFICAS .....</b>	<b>97</b>
<b>APÊNDICE A – FORMULAÇÃO DO PROBLEMA DE CAIXEIRO VIAJANTE.....</b>	<b>99</b>
<b>ANEXO A – DADOS DO EXEMPLO DO ESTUDO DE CASO .....</b>	<b>103</b>
<b>ANEXO B – CÓDIGO EM VBA DO MODELO DE ROTEIRIZAÇÃO .....</b>	<b>111</b>

## LISTA DE TABELAS

<i>Tabela 3.2.1 – Comparaçao entre o método de varredura e a heurísitca de Clarke &amp; Wright</i>	44
<hr/>	
<i>Tabela 3.3.1 – Demanda e localização dos clientes</i>	54
<i>Tabela 3.3.2 – Matriz de distâncias entre os pontos do problema</i>	55
<i>Tabela 3.3.3 – Solução inicial do exemplo reduzido</i>	56
<i>Tabela 3.3.4 – Resumo das iterações da resolução exata do exemplo reduzido</i>	57
<i>Tabela 3.3.5 – Frações de demanda atendidas pelos veículos</i>	58
<i>Tabela 3.3.6 – Reclassificação dos pontos em ordem crescente de ângulo de coordenada polar</i>	59
<i>Tabela 3.3.7 – Demanda acumulada dos clientes e alocação dos veículos</i>	61
<i>Tabela 3.3.8 - Rotas dos veículos após a primeira varredura</i>	62
<i>Tabela 3.3.9 – Resultados do método de varredura</i>	63
<i>Tabela 3.3.10 – Comparaçao das soluções obtidas com a situação real</i>	64
<i>Tabela 4.2.1 – Informações gerais sobre os veículos</i>	66
<i>Tabela 4.2.2 – Fatores médios de circuito em alguns países – Adaptado de Ballou (2005)</i>	69
<i>Tabela 4.2.3 – Dados gerais sobre a distribuição</i>	70
<i>Tabela 4.3.1 – Compilação dos dados de entrega</i>	71
<i>Tabela 6.1.1 – Resultados obtidos na simulação com 25 pontos</i>	82
<i>Tabela 6.2.1 – Resultados das simulações do modelo</i>	83
<i>Tabela 6.2.2 – Economias geradas em relação à situação real</i>	86
<i>Tabela 6.2.3 – Comparaçao entre as soluções obtidas com as soluções ótimas</i>	92



## LISTA DE FIGURAS

<i>Figura 1.1.1 – Fluxograma das operações da empresa.....</i>	12
<i>Figura 1.2.1 – Fluxograma do processo atual de distribuição.....</i>	16
<i>Figura 1.3.1 – Exemplo ilustrativo do problema de roteirização de veículos com entregas fracionadas – Extraído de Dror, Laporte e Trudeau (1994).....</i>	18
<i>Figura 2.1.1 – Solução de um problema de roteirização de veículos sem entregas fracionadas – Extraído de Chen et al. (2006) .....</i>	23
<i>Figura 2.1.2 – Solução de um problema de roteirização de veículos com entregas fracionadas– Extraído de Chen et al. (2006) .....</i>	23
<i>Figura 2.2.1 – Conservação dos fluxos de entrada e saída .....</i>	28
<i>Figura 2.2.2 – Todas as demandas dos clientes serão atendidas, sendo permitidas entregas fracionadas .....</i>	29
<i>Figura 2.2.3 – Se um cliente <math>i</math> é atendido por um veículo <math>v</math>, haverá obrigatoriamente um arco que sai dele .....</i>	30
<i>Figura 2.2.4 – Configuração não permitida pela restrição (2.18) .....</i>	31
<i>Figura 2.2.5 – Formação de uma subrota no grafo <math>G(N,A)</math> .....</i>	32
<i>Figura 2.2.6 –Contra-exemplo mostrando que a restrição (2.19) não é válida para o SDVRP .....</i>	33
<i>Figura 3.1.1 – Exemplo de aplicação do processo iterativo de eliminação de subrotas – primeira iteração.....</i>	38
<i>Figura 3.1.2 – Exemplo de aplicação do processo iterativo de eliminação de subrotas – segunda iteração.....</i>	39
<i>Figura 3.1.3 – Exemplo de aplicação do processo iterativo de eliminação de subrotas – terceira iteração .....</i>	40
<i>Figura 3.1.4 – Exemplo de aplicação do processo iterativo de eliminação de subrotas – quarta iteração .....</i>	41
<i>Figura 3.1.5 – Exemplo de aplicação do processo iterativo de eliminação de subrotas – solução ótima.....</i>	42
<i>Figura 3.2.1 – Classificação dos ângulos de coordenadas polares em ordem crescente.....</i>	47
<i>Figura 3.2.2 – Roteiros formados com o método da varredura.....</i>	48
<i>Figura 3.2.3 – Rotação do eixo <math>(x,y)</math> no sentido anti-horário .....</i>	49
<i>Figura 3.2.4 – Adaptação do passo de rotação do eixo <math>(x,y)</math>.....</i>	50

<i>Figura 3.2.5 – O ponto i não pode ser atendido pelo veículo 1 no modelo original do método de varredura.....</i>	51
<i>Figura 3.2.6 – Adaptação do método de varredura .....</i>	52
<i>Figura 3.3.1 – Disposição geográfica dos pontos.....</i>	54
<i>Figura 3.3.2 – Início do procedimento de varredura .....</i>	60
<i>Figura 3.3.3 – Rotação do eixo (x,y) para execução da segunda varredura .....</i>	62
<i>Figura 3.3.4 – Solução obtida através do método de varredura .....</i>	63
<i>Figura 5.1.1 – Descrição do algoritmo .....</i>	77
<i>Figura 6.2.1 – Comparação entre as soluções obtidas e a situação real.....</i>	85
<i>Figura 6.2.2 – Economias geradas com a aplicação do modelo.....</i>	87
<i>Figura 6.2.3 – Percentual das economias geradas .....</i>	88
<i>Figura 6.2.4 – Desempenho computacional do algoritmo nos problemas simulados.....</i>	89
<i>Figura 6.2.5 – Tempo médio de computação por classe de problema (A, B e C) .....</i>	90
<i>Figura 6.2.6 – Variação dos tempos computacionais com o número de pontos .....</i>	91

## INTRODUÇÃO

Logística empresarial é um termo que vem recebendo grande destaque no mundo corporativo ao longo dos últimos anos. Tal fenômeno se deve em grande parte às significativas transformações econômicas das últimas décadas, as quais influenciaram fortemente o modelo de gestão das empresas ao redor do mundo. Em decorrência dessas mudanças, a crescente concorrência entre as empresas passou a se dar em âmbito global.

A logística passou a integrar esse contexto como uma das competências vitais para a gestão de negócios, agregando importante valor aos clientes e transformando-se em uma importante fonte de vantagem competitiva. A importância da logística e da cadeia de suprimentos se observa na criação de valor – valor tanto para os clientes quanto para os fornecedores da empresa, além de todos os diretamente interessados. O valor da logística se manifesta basicamente em termos de tempo e lugar, ou seja, disponibilidade. Bens e serviços não têm valor, a menos que estejam em poder dos clientes quando (tempo) e onde (lugar) eles pretendem consumi-los. Uma administração logística adequada compreende que cada atividade na cadeia de suprimentos contribui para o processo de agregação de valor (BALLOU, 2005).

As decisões de transporte são parte fundamental da estratégia e planejamento logístico, com destaque para o problema de roteirização e programação de veículos. Normalmente, o transporte representa de um a dois terços dos custos logísticos totais (BALLOU, 2005); e dessa forma, a utilização eficiente dos equipamentos e pessoal de transporte é uma importante fonte de redução e otimização de custos. Com isso, a determinação dos melhores roteiros para os veículos, a fim de minimizar os tempos de viagem e distâncias percorridas, gera importantes benefícios para empresas em qualquer ramo de atividade, em termos de redução de custos e melhoria no nível de serviço ao cliente (MIURA, 2003).

Particularmente, no caso de uma empresa importadora, cujas atividades dependem fortemente das operações logísticas para funcionar, o problema de roteirização de veículos é de extrema importância, principalmente na distribuição dos produtos aos canais varejistas. Nesse contexto, o presente trabalho propõe um modelo de roteirização de veículos para uma empresa importadora de produtos do Japão, TRADBRAS IMPORTAÇÃO E EXPORTAÇÃO, com foco na roteirização dos veículos que atuam na distribuição dos produtos importados aos canais varejistas da cidade de São Paulo.

Basicamente, o objetivo do trabalho consiste na determinação da melhor maneira de alocar os veículos aos diversos estabelecimentos de varejo, programando o melhor roteiro diário, de maneira a minimizar os custos de transporte e o tempo total de viagem, garantindo o atendimento da demanda de todas as lojas e satisfazendo todas as restrições impostas pelo problema. Dessa forma, a proposta de solução do problema envolverá extensivamente a aplicação de técnicas e ferramentas de Pesquisa Operacional, voltadas para a resolução de problemas de roteirização e programação de veículos.

O trabalho está dividido em sete capítulos, de acordo com seguinte estrutura lógica.

O capítulo 1 consiste na apresentação e descrição da empresa estudada, a TRADBRAS IMPORTAÇÃO E EXPORTAÇÃO, contendo um panorama geral da empresa e do mercado de produtos orientais, além de uma descrição das principais atividades e processos, em especial a distribuição dos produtos ao varejo. Na seqüência deste capítulo o problema abordado, ou seja, o problema de roterização de veículos será definido e contextualizado na situação atual da empresa.

O capítulo 2 consiste em uma revisão da literatura dos principais conceitos envolvendo o problema de roteirização de veículos, em especial o problema de roteirização com entregas fracionadas, compondo um referencial teórico que guiará o desenvolvimento do presente trabalho.

O capítulo 3 descreve o método de resolução a ser utilizado no problema definido, com base no referencial teórico mostrado no capítulo anterior, apresentando a metodologia sobre a qual se fundamentará o modelo a ser desenvolvido. Um exemplo prático, ilustrando uma situação real ocorrida na empresa será apresentado como forma de demonstrar a dinâmica de funcionamento do modelo de resolução proposto.

O capítulo 4 dedica-se ao levantamento e coleta dos dados relevantes e necessários para a resolução do problema de roteirização abordado neste estudo. Serão apresentados os dados necessários para entrada no modelo proposto, bem como a metodologia utilizada no levantamento dos mesmos, de acordo com os parâmetros e características do problema de roterização de veículos com entregas fracionadas.

O capítulo 5 destina-se à etapa de implantação e validação do modelo proposto, como forma de analisar sua eficiência e agilidade na resolução do problema abordado,

possibilitando a aplicação prática do modelo nas operações da empresa. O processo de modelagem do algoritmo a ser utilizado para a resolução do problema, bem como os testes necessários para a validação do modelo, serão apresentados neste capítulo.

O capítulo 6 apresenta a análise dos resultados obtidos com a aplicação do modelo em situações práticas do cotidiano da empresa. Tal análise possibilitará a avaliação do modelo proposto tanto em termos de eficiência computacional quanto em qualidade da solução obtida em relação à situação atual (real) que ocorre na prática, norteando o desenvolvimento das conclusões do estudo.

O capítulo 7 apresenta as conclusões do presente trabalho. Basicamente, trata-se de um fechamento do estudo realizado, com as principais questões abordadas e uma análise crítica explorando os pontos fortes e deficiências do modelo proposto, servindo como ponto de partida para o desenvolvimento de trabalhos futuros.



## **1. DESCRIÇÃO DA EMPRESA E DEFINIÇÃO DO PROBLEMA**

Este capítulo tem como finalidade a contextualização da empresa estudada para a definição do problema e do objetivo do trabalho. A parte inicial do capítulo consiste na apresentação da empresa na qual o trabalho se desenvolverá, TRADBRAS IMPORTAÇÃO E EXPORTAÇÃO, doravante denominada TRADBRAS, contendo uma visão geral da empresa, além de suas principais operações e processos. A segunda parte é dedicada à definição do problema que fundamentará o trabalho, assim como os objetivos pretendidos com a sua resolução.

### **1.1. DESCRIÇÃO DA EMPRESA**

A TRADBRAS é uma empresa que atua no ramo de importação de produtos japoneses desde 1966. A empresa sempre buscou pautar sua filosofia de trabalho nos pilares e ensinamentos da cultura milenar japonesa, baseando-se em três conceitos fundamentais: responsabilidade, compromisso e tradição. O seguimento desses princípios no dia-a-dia de trabalho se traduz na credibilidade da empresa com seus clientes e no atendimento diferenciado, elevando a empresa ao patamar de uma das mais tradicionais e importantes do ramo.

A TRADBRAS é uma empresa familiar de médio porte sediada em São Paulo, com operações em todo o Brasil, principalmente nos estados de São Paulo, Paraná, Rio de Janeiro, Mato Grosso do Sul. Entretanto, as principais atividades se concentram no estado de São Paulo, o principal mercado consumidor de produtos orientais no país.

#### **O mercado de produtos orientais**

O ambiente econômico das últimas décadas, inclusive com a introdução do Plano Real e a conseqüente queda na inflação, proporcionou um aumento significativo no poder de compra da população brasileira (CUNHA, 2001). Este cenário positivo favoreceu enormemente o desenvolvimento do mercado de importados no país. A partir desse período, observou-se um crescimento notável nos negócios relacionados à importação, como foi o caso da TRADBRAS no final da década de 90. Essa expansão mostra o bom momento do mercado de produtos orientais no país, que, com a comemoração do centenário da imigração japonesa no ano de 2008, estará ainda mais aquecido no biênio 2007/2008.

O mercado brasileiro de produtos orientais é predominantemente dominado por empresas que seguem o perfil da TRADBRAS, ou seja, empresas familiares e de médio porte, onde a tradição do negócio é fator preponderante. Por ser uma empresa tradicionalmente reconhecida no ramo de produtos orientais, a TRADBRAS ocupa uma posição de destaque (principalmente no eixo Sul-sudeste), atendendo os maiores e mais importantes clientes desse mercado. Os principais produtos e clientes atendidos pela TRADBRAS são apresentados a seguir.

### **Principais produtos e clientes**

A empresa conta com mais de 1.500 SKU comercializados e distribuídos, divididos em bebidas, alimentos, utilidades domésticas diversas e artigos para pesca, com especial destaque para o saquê *Hakushika*, muito famoso no Japão. Trata-se do produto individual que apresenta a maior margem de contribuição para a empresa, embora não seja o produto com o maior volume de vendas. A TRADBRAS apóia-se fortemente sob a marca *Hakushika*, uma vez que a empresa é a sua única representante no Brasil. A força desta marca, juntamente à sua associação com a TRADBRAS constitui um importante fator competitivo para a empresa.

A seguir serão apresentadas as categorias de produtos comercializados pela TRADBRAS, a saber: alimentos, bebidas, utilidades domésticas e artigos de pesca, com exemplos dos principais produtos de cada categoria.

#### **• Alimentos**

Esta categoria compreende alimentos industrializados diversos, perecíveis e não-perecíveis tais como macarrão instantâneo, salgadinhos, chocolates, biscoitos, balas e outros tipos de doces, molho de soja, algas marinhas torradas, conservas, entre outros. A categoria de alimentos representa a maior parcela das vendas totais, no entanto em termos de produtos individuais, o saquê *Hakushika* (classificado na categoria de bebidas) é o item que apresenta a maior margem contribuição da empresa, conforme mencionado anteriormente.

#### **• Bebidas**

Esta categoria inclui bebidas diversas, alcoólicas e não-alcoólicas, dentre as quais se destacam: cervejas, chás e saquês. A empresa comercializa bebidas envasadas tanto em garrafas (plásticas e de vidro) quanto em latas de alumínio. A categoria de bebidas é a que

apresenta a segunda maior contribuição para o faturamento global da empresa. As vendas do saquê *Hakushika* correspondem a cerca de 80% do total das vendas de bebidas.

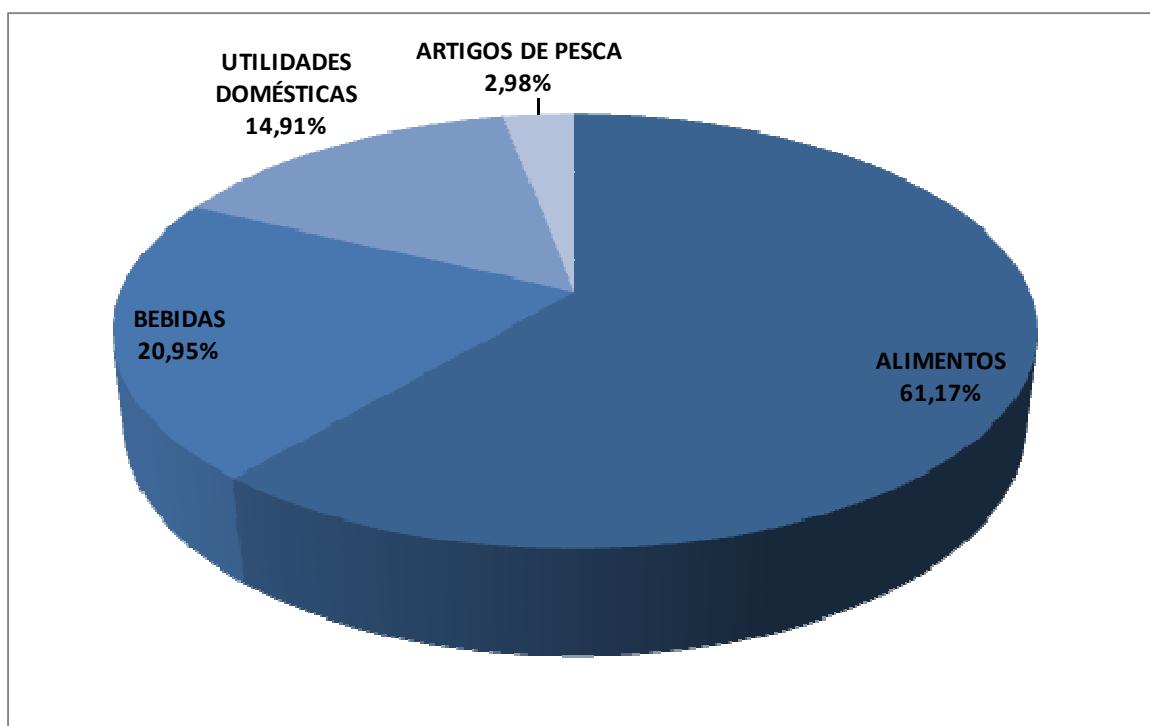
- **Utilidades domésticas**

Esta categoria é composta basicamente por utensílios de cozinha, como panelas, facas de cozinha, talheres, potes, embalagens, panelas elétricas, grelhas, copos para chá, tigelas, entre outros.

- **Artigos de pesca**

Esta categoria é composta de artigos de pesca em geral: iscas artificiais, anzóis, linhas de pesca, carretilhas, entre outros.

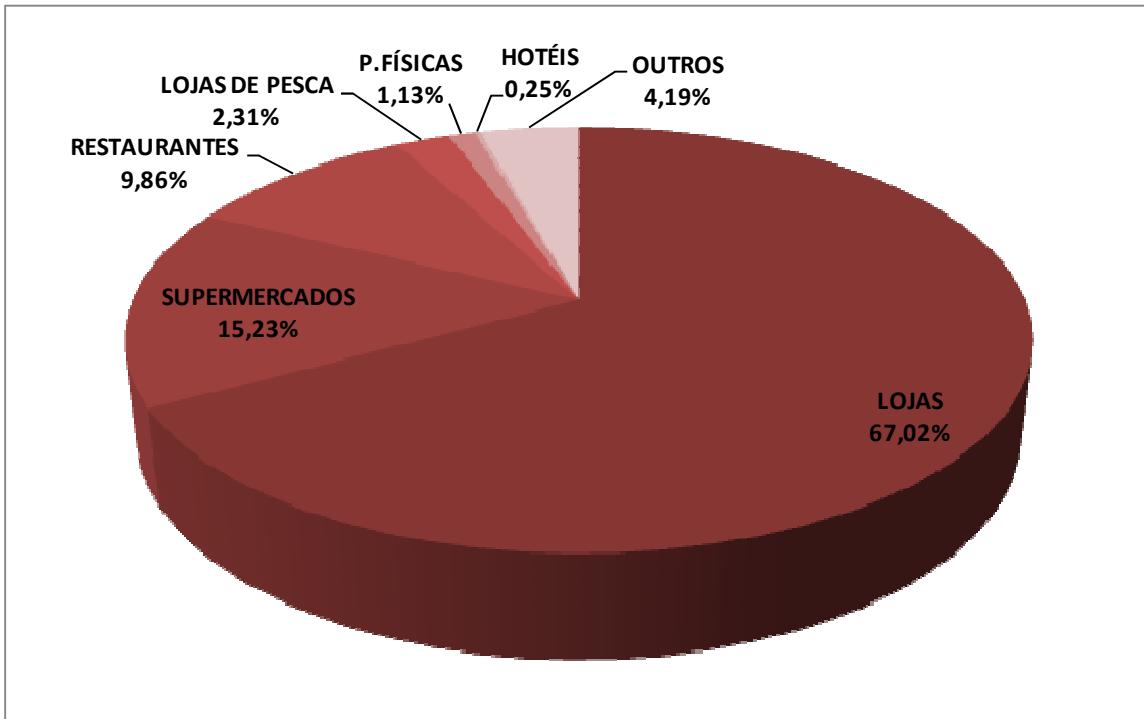
O gráfico 1.1.1 a seguir mostra a participação relativa das categorias de produtos em termos de faturamento.



*Gráfico 1.1.1 – Participação no faturamento por categoria de produto*

A TRADBRAS possui em seu cadastro mais de 1.000 clientes ativos, dentre os quais se destacam: lojas de conveniência especializadas em produtos orientais, redes de supermercados, restaurantes de cozinha japonesa e hotéis, além de pessoas físicas. Em São

Paulo, a empresa conta com mais de 500 clientes distribuídos por toda a cidade. O gráfico 1.1.2 a seguir mostra a participação relativa de cada segmento no total de vendas



*Gráfico 1.1.2 – Participação no faturamento por segmento*

### **1.1.1. Descrição das operações da empresa**

Basicamente, as operações da TRADBRAS podem ser subdivididas em dois macro-processos: o processo de importação e a distribuição dos produtos ao varejo.

#### **O processo de importação**

De maneira geral, três entidades atuam diretamente no processo de importação: a empresa (TRADBRAS), os exportadores e os fornecedores.

Os exportadores são os responsáveis pelos pedidos vindos do Brasil, realizados pela TRADBRAS. Após o recebimento dos pedidos da empresa, os exportadores ficam encarregados de efetuar as compras dos produtos junto aos fornecedores no Japão nas quantidades acordadas, no melhor preço possível. Fisicamente, os exportadores se localizam em território japonês.

Como a empresa possui mais de 1.500 SKU, é inviável a negociação com cada um dos diversos fornecedores individualmente. Neste sentido, os exportadores atuam na intermediação entre a empresa e os fornecedores, facilitando o processo de compras para a TRADBRAS. No entanto, há alguns casos em que a empresa tem contato direto na negociação com os fornecedores.

## O processo de distribuição

O processo de distribuição consiste basicamente na entrega dos produtos importados junto aos clientes. A distribuição pode ocorrer tanto com a interferência direta da empresa, no caso das entregas locais (São Paulo) quanto através de transportadoras no caso das entregas em outros estados.

A seguir, a dinâmica das operações da empresa é mostrada em maior detalhe.

### 1. Pedido de mercadorias

Quando o estoque de mercadorias atinge um nível mínimo (estoque para 70 a 80 dias), a empresa negocia os pedidos junto aos exportadores no Japão. A empresa então monta contêineres com os produtos, otimizando os custos de frete marítimo, evitando a alocação de produtos volumosos de baixo valor agregado e buscando a máxima utilização da capacidade de cada contêiner, dados os seus altos custos fixos e de manuseio.

### 2. Ordem de pedido para os exportadores

Após a montagem do contêiner, a empresa envia a ordem de pedido ao exportador no Japão. O exportador então efetua o pedido junto aos fornecedores e envia um documento à TRADBRAS contendo as quantidades, peso, dimensões (volume), preços e descrição dos itens a serem despachados no contêiner, bem como o preço total do contêiner. Este documento é denominado *invoice*.

### 3. Fechamento de pedido junto aos exportadores

O *invoice* enviado pelos exportadores é analisado pela TRADBRAS, que confere se os preços e quantidades estão corretos de acordo com a configuração do contêiner previamente determinada. Caso haja alguma não-conformidade, a empresa solicita as correções necessárias. Quando os dados do *invoice* estão totalmente corretos, o pedido é fechado junto à exportadora.

#### **4. Despacho do contêiner via transporte marítimo**

Após o fechamento e consolidação do pedido, a exportadora providencia o carregamento e o embarque do contêiner rumo ao Brasil. O tempo de viagem entre os portos no Japão e o porto de Santos é de aproximadamente 60 dias. Como grande parte desse estoque em trânsito é perecível, a empresa toma um cuidado especial no sentido de não permitir atrasos na entrega dos produtos, além de exigir medidas de segurança no manuseio e transporte dos produtos.

#### **5. Desembarque do contêiner no Brasil**

O contêiner transportado é então desembarcado no porto de Santos, e fica retido nas docas até a liberação aduaneira.

#### **6. Preparação da documentação pelo despachante para liberação**

A empresa, através de um despachante preenche toda a documentação aduaneira e de importação e recolhe todos os tributos, liberando assim o contêiner

#### **7. Retirada do contêiner e entrega na sede da TRADBRAS**

O transporte do contêiner até o armazém da empresa é realizado por uma transportadora contratada, através de caminhões. O contêiner é descarregado no armazém e os paletes são transportados até as prateleiras, onde são acondicionados. As prateleiras são divididas em categorias de produtos.

#### **8. Distribuição das mercadorias para o varejo**

A distribuição das mercadorias para o varejo ocorre de duas maneiras distintas: uma para as entregas realizadas na cidade de São Paulo e outra para as entregas nas demais localidades.

Atualmente, a distribuição das mercadorias (i.e., os pedidos dos clientes) na cidade de São Paulo é terceirizada, realizada por uma empresa que realiza as entregas através de *vans*. Cabe ressaltar que na situação presente, nenhum método formal de roteirização de veículos é empregado, e os roteiros são determinados pela empresa apenas com base nas necessidades da demanda e bom-senso.

As entregas nas demais localidades são realizadas por transportadoras contratadas e ocorrem duas situações diferentes, dependendo do contrato firmado: na primeira, a empresa se encarrega de levar os pedidos até o CD de uma das distribuidoras; na segunda, a própria transportadora realiza a coleta no armazém da empresa. Vale ressaltar que isso ocorre porque não há um contrato fixado com uma única transportadora. Além disso, uma vez que os pedidos passam para a transportadora, o transporte até os destinos é de sua inteira responsabilidade e nesse caso o frete é pago pelo cliente (preço CIF – *Cost, Insurance and Freight*).

A grande maioria das entregas é realizada via modal rodoviário. Em raras ocasiões é utilizado o modal aéreo.

O fluxograma da figura 1.1.1 a seguir ilustra o processo como um todo.

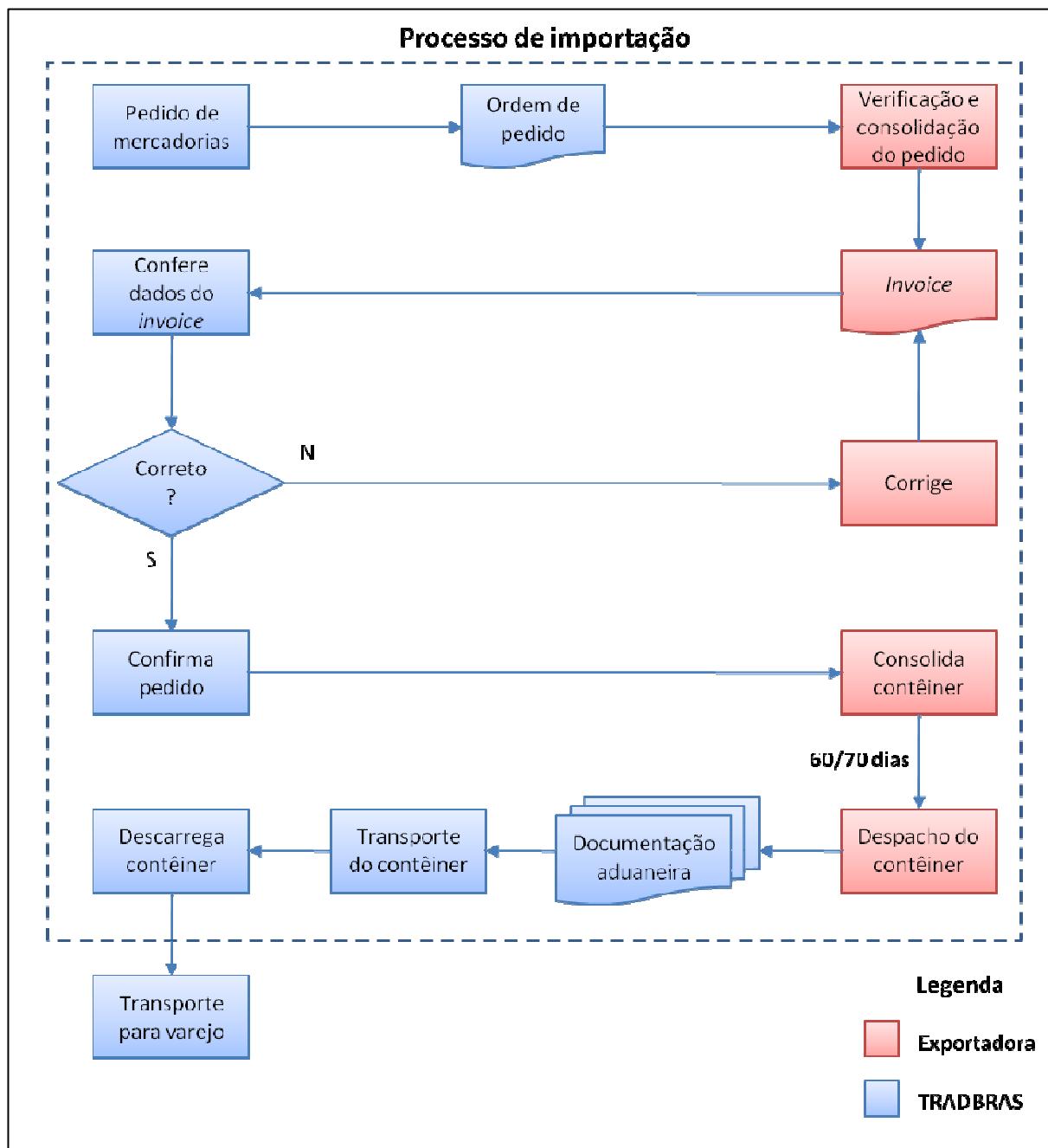


Figura 1.1.1 – Fluxograma das operações da empresa

### **1.1.2. Vínculo com a empresa**

O contato e relacionamento com a organização se dão por meio de um vínculo familiar muito próximo com o proprietário da empresa, permitindo o conhecimento profundo das operações e processos realizados, bem como a gestão do negócio em si. Este contato proporciona uma considerável facilidade de obtenção de dados e agendamento de visitas, viabilizando, portanto a realização do trabalho, mesmo sem a existência de um vínculo de estágio entre a empresa e o autor.

Além disso, a facilidade de acesso à empresa permitiu a pronta identificação do problema, tratando-se dessa forma, de uma boa oportunidade em propor e desenvolver um modelo de resolução utilizando as técnicas e ferramentas abordadas no curso de Engenharia de Produção, em especial na área de Pesquisa Operacional e Logística.

## **1.2. DEFINIÇÃO DO PROBLEMA**

O presente trabalho trata de um problema de decisões de transportes relacionadas à roteirização de veículos e consolidação de cargas, a partir de um depósito central, ao menor custo possível, respeitando as restrições de demanda dos clientes, jornadas de trabalho dos motoristas e capacidade dos veículos. A seleção desse tema se justifica por uma deficiência identificada na empresa, que é a ausência de um método formal na roteirização dos veículos que realizam as entregas dos pedidos para os restaurantes e estabelecimentos de varejo na cidade de São Paulo.

Conforme mencionado no item 1.1.1 – *Descrição das operações da empresa*, a entrega dos pedidos é realizada por uma empresa terceirizada, através de *vans*. Esta empresa contratada apenas fornece os veículos e motoristas para uso da TRADBRAS; os roteiros são determinados pela própria empresa.

No entanto, os procedimentos utilizados na determinação dos roteiros são informais e realizados sem critérios específicos. Basicamente, o principal critério utilizado é a divisão das entregas por região da cidade, porém sem a formação de roteiros em âmbito micro, ou seja, específicos para uma dada região. Dessa forma, uma vez que um veículo é designado para determinada região, não há um roteiro específico a seguir dentro dela.

Portanto, é freqüente a ocorrência de um grande número de passagens a uma mesma região, ou seja, em muitos casos o veículo passa diversas vezes pela mesma rua ou local

desnecessariamente. Em decorrência disso, as viagens são mais longas do que o necessário, tanto em termos de distância percorrida, quanto em relação aos tempos totais de viagem.

Outro fator que agrava a determinação formal de roteiros para a empresa é o elevado número de clientes espalhados pela cidade. Esta demanda pulverizada torna muito difícil por parte da empresa a determinação de roteiros específicos para os veículos, uma vez que não há na empresa um departamento especializado em planejamento logístico, nem profissionais dedicados para tanto no quadro de funcionários.

Para uma melhor visualização do que o problema representa, o processo atual de distribuição na cidade de São Paulo será explicitado em maiores detalhes.

### **1.2.1. O processo atual de distribuição**

Na configuração atual, a empresa conta com duas *vans* à disposição fornecidas pela contratada, incluindo os motoristas e ajudantes. A empresa então fica encarregada de estabelecer os roteiros ao longo dos dias da semana. Assim que os clientes efetuam os pedidos, a empresa possui uma política interna de se comprometer a entregá-los em um prazo de até 48 horas após a sua realização.

As entregas podem ser classificadas em três categorias distintas, de acordo com o porte do cliente e natureza da demanda: as entregas para estabelecimentos de grande porte, estabelecimentos de médio e pequeno porte e restaurantes. Os estabelecimentos de grande porte se caracterizam por realizar grandes pedidos (inclusive existem casos em quantidades demandadas por um único cliente excedem a capacidade de carga dos veículos), com quantidades aproximadamente fixas e em intervalos de tempo regulares.

Já os estabelecimentos de médio e pequeno porte, realizam pedidos de dimensões menores, ou seja, que não ultrapassam a capacidade dos veículos, apresentando uma variação um pouco maior na demanda, porém não a ponto de ser considerada estocástica.

Finalmente, os restaurantes apresentam um perfil de demanda bastante variável, tanto em termos de periodicidade de realização dos pedidos, quanto de quantidades e produtos pedidos, embora em geral, os pedidos são pequenos (poucas quantidades).

Apesar da grande diversidade de produtos entregues, as cargas são modularizadas, ou seja, os pedidos são acondicionados em caixas de tamanho padrão, para serem alocadas no interior dos veículos.

Depois do recebimento e verificação dos pedidos, a empresa levanta quais clientes serão atendidos a cada dia, de acordo com suas demandas e prioridades e então despacha os veículos para realizar as entregas. O processo de entregas se dá em base diária. Entretanto, uma vez que o veículo é despachado para os clientes pré-determinados, fica a critério do motorista o roteiro a ser seguido. O que se observa, portanto, são elevados tempos de viagem e passagens desnecessárias de um veículo em uma mesma rua ou região.

Em alguns casos, os dois veículos disponíveis não conseguem atender todos os clientes do dia em uma única viagem e por isso retornam ao depósito da TRADBRAS para o carregamento de mais mercadorias. No entanto, em muitos desses retornos o veículo é carregado com menos da metade de sua capacidade.

Os motoristas trabalham em turnos de 8 horas diárias (início às 09:00 e término às 17:00), de segunda a sexta. No entanto, em termos práticos há uma grande variabilidade nas jornadas de trabalho, uma vez que, em determinados dias eles terminam as entregas em uma região antes do término do turno, havendo consequentemente, um considerável tempo ocioso. Inversamente, em outros casos ocorre o oposto; os motoristas ficam sobrecarregados e mal conseguem terminar as entregas dentro da jornada diária.

O fluxograma da figura 1.2.1 mostra o processo de distribuição em maiores detalhes.

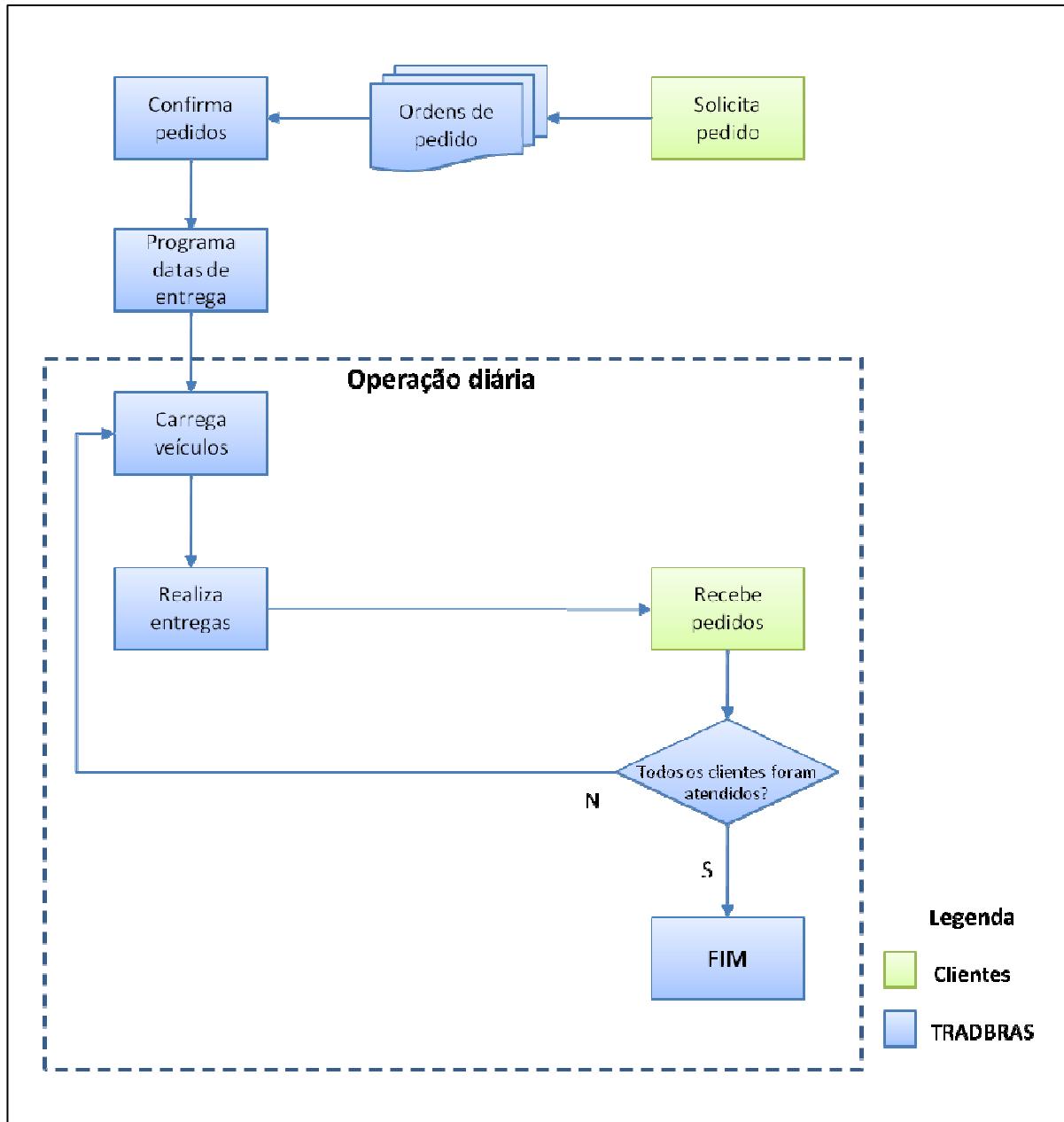


Figura 1.2.1 – Fluxograma do processo atual de distribuição

### 1.3. OBJETIVO GERAL DO TRABALHO

O problema apresentado na seção anterior mostra-se como uma grande oportunidade para a aplicação de técnicas de Pesquisa Operacional, como forma de otimizar os roteiros de entrega realizados pelos veículos. O problema abordado traz impactos altamente negativos para a empresa, uma vez que a falta de métodos científicos e modelos para a determinação dos roteiros acarreta em um grande e muitas vezes desnecessário número de viagens, e consequentemente, custos elevados de transporte são incorridos. Além disso, a inadequada

programação ocasiona atrasos indesejáveis nas entregas, o que é altamente prejudicial para a credibilidade da empresa no mercado.

Além da subjetividade e da falta de rigor nos critérios, fatores como o aproveitamento da capacidade dos veículos e a jornada de trabalho dos motoristas e ajudantes, não são considerados adequadamente.

O problema abordado é uma instância do problema de roteirização de veículos com entregas fracionadas (SDVRP – *Split Delivery Vehicle Routing Problem*). Nesta variação do problema tradicional de roteirização, um ponto de demanda pode ser atendido por mais de um veículo, onde a demanda de cada ponto pode ser maior ou menor que a capacidade dos veículos (BELFIORE, 2006). A aplicação do problema ao caso da empresa estudada apresenta as características mostradas a seguir.

Os veículos partem de um depósito central, no caso, o armazém da TRADBRAS, distribuindo diversos tipos de produtos, atendendo clientes durante a jornada de trabalho dos motoristas. A frota de veículos é homogênea, com capacidade fixa  $C$ . Em um determinado dia, a demanda de um cliente  $i$  é dada por  $q_i$ . Ao final da jornada ou quando todos os produtos forem entregues, o veículo retorna ao armazém. Além disso, um dado ponto de demanda pode ser atendido por mais de um veículo.

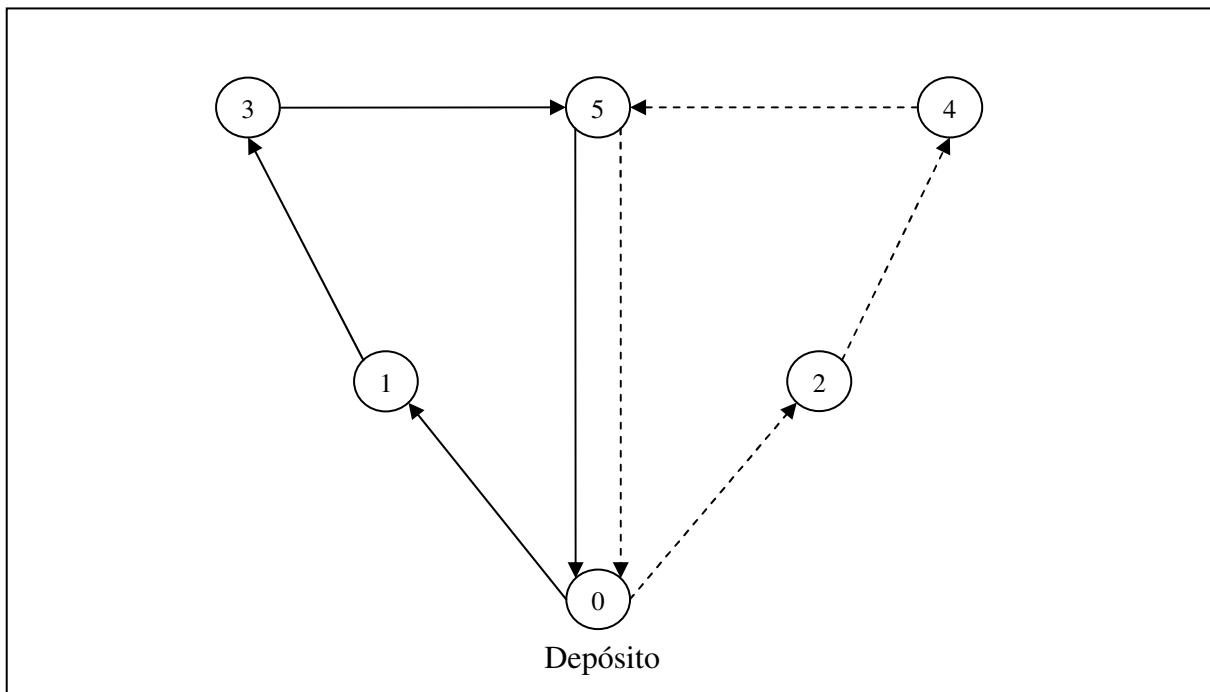
Sendo assim, para uma dada configuração de pedidos de clientes (local e quantidade), o objetivo do modelo se traduz em determinar roteiros ótimos a serem percorridos pelos veículos e, no caso das entregas para clientes cuja demanda seja maior que a capacidade dos veículos, determinar a quantidade a ser alocada para cada veículo, visando à minimização dos custos de transporte, tempo de entrega e número de viagens realizadas. Além disso, o modelo deve respeitar as seguintes restrições: atendimento da demanda de todos os clientes e a capacidade máxima de carga dos veículos.

As principais hipóteses adotadas no problema são:

- A demanda dos clientes é determinística;
- A unidade de pedidos é medida em termos de caixas, uma vez que os produtos são acondicionados em caixas de tamanho padrão antes de serem carregadas nos veículos;
- Um cliente individual pode ser atendido por mais de um veículo;

- Os roteiros iniciam e terminam no depósito;
- As jornadas de trabalho são flexíveis, ou seja, não há roteiros incompletos devido ao término da jornada de trabalho dos motoristas e ajudantes.

A figura 1.3.1 a seguir, extraída de Dror, Laporte e Trudeau (1994) apresenta um exemplo ilustrativo do problema de roteirização de veículos com entregas fracionadas (SDVRP). No grafo, o nó de índice 0 (zero) representa o depósito, a linha contínua representa o roteiro do veículo 1 e a linha tracejada representa o trajeto do veículo 2.



*Figura 1.3.1 – Exemplo ilustrativo do problema de roteirização de veículos com entregas fracionadas –  
Extraído de Dror, Laporte e Trudeau (1994)*

Conforme pode ser observado na figura 1.3.1, no problema de roteirização com entregas fracionadas, é permitido o atendimento de um cliente por mais de um veículo. No exemplo, a demanda do cliente 5 é dividida entre os dois veículos. Esta relaxação do problema clássico, onde a demanda de cada cliente somente pode ser atendida por um único veículo, permite uma melhor utilização da capacidade dos veículos, proporcionando economias tanto em termos de custos (distâncias), quanto ao número de veículos utilizado (DROR; TRUDEAU, 1990).

## 2. REVISÃO DA LITERATURA

Este capítulo tem como objetivo apresentar o referencial teórico necessário para o desenvolvimento do presente trabalho, sendo de fundamental importância para o entendimento do modelo de resolução proposto, norteando o desdobramento do tema do trabalho e a subsequente definição da metodologia de resolução do problema. Para tanto, serão descritos os principais conceitos encontrados na literatura, envolvendo o problema de roteirização de veículos, em especial o caso de entregas fracionadas.

Inicialmente, será apresentada uma breve definição do problema clássico de roteirização de veículos, como apresentado na formulação de Fisher e Jaikumar (1981) e o caso particular do problema de roteirização com entregas fracionadas (SDVRP – *Split Delivery Vehicle Routing Problem*), introduzido na literatura por Dror e Trudeau (1989). A segunda parte deste capítulo consiste em uma discussão acerca da complexidade computacional do problema de roteirização de veículos com entregas fracionadas. Finalmente, a terceira parte deste capítulo apresenta em detalhes a formulação matemática do SDVRP.

### 2.1. PROBLEMAS DE ROTEIRIZAÇÃO DE VEÍCULOS

Goldbarg e Luna (2002) definem a idéia básica do problema de roteirização como: “com o uso de veículos, visitar uma série de clientes ao menor custo possível, atendendo todas as demais imposições do problema”.

O problema de roteirização de veículos em sua forma mais simples é definido como um problema de distribuição no qual os veículos devem ser programados para atender clientes geograficamente dispersos e de demanda conhecida, partindo de um depósito central. As restrições mais comuns do problema estão associadas à capacidade dos veículos (CHRISTOFIDES, 1985). Trata-se então de um problema de múltiplos caixeiros viajantes, com capacidade limitada, donde se observa que os métodos de resolução de problemas de roteirização de veículos estão fortemente relacionados ao problema clássico do caixeiro viajante (GOLDBARG; LUNA, 2000), tanto pelas similaridades na conceituação do problema, quanto pelo forte caráter combinatório.

Os problemas de roteirização de veículos podem ser classificados em várias categorias e tipos. Os diversos problemas diferem entre si de acordo com aspectos relacionados ao tipo de operação, tipo de carga, tipo de frota utilizada, à localização dos clientes, à natureza das

restrições, ao tipo de função objetivo, entre outros (BELFIORE, 2006). No entanto, grande parte dos problemas de roteirização de veículos, entre eles o problema de roteirização de veículos com entrega fracionada, derivam diretamente do problema clássico.

### 2.1.1. Formulação do problema clássico de roteirização de veículos

Uma formulação básica para o problema clássico de roteirização de veículos foi introduzida por Fisher e Jaikumar (1981), na publicação “*A Generalized Assignment Heuristic for Vehicle Routing*” e será apresentada resumidamente a seguir, pois apresenta características semelhantes à formulação do SDVRP.

#### *Parâmetros*

- $NV$  = Número de veículos
- $n$  = Número de clientes para os quais uma entrega deve ser feita. Os clientes são indexados de 1 a  $n$  e o índice 0 representa o depósito central.
- $Q_v$  = Capacidade do veículo  $v$
- $q_i$  = Demanda do cliente  $i$
- $c_{ij}$  = Custo de viagem do cliente  $i$  para o cliente  $j$
- $S$  = Representa um subgrafo qualquer do problema, excluindo o depósito

#### *Variáveis*

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

$$x_{ij}^v = \begin{cases} 1, & \text{se o veículo } v \text{ viaja diretamente do cliente } i \text{ para o cliente } j \\ 0, & \text{caso contrário} \end{cases}$$

#### *Formulação*

$$\min \sum_i \sum_j \sum_v c_{ij} x_{ij}^v \quad (2.1)$$

s.a.

$$\sum_i q_i y_i^v \leq Q_v \quad v = 1, \dots, NV \quad (2.2)$$

$$\sum_v y_0^v = NV \quad (2.3)$$

$$\sum_v y_i^v = 1 \quad i = 1, \dots, n \quad (2.4)$$

$$\sum_i x_{ij}^v = y_j^v \quad j = 1, \dots, n ; v = 1, \dots, NV \quad (2.5)$$

$$\sum_j x_{ij}^v = y_i^v \quad i = 1, \dots, n ; v = 1, \dots, NV \quad (2.6)$$

$$\sum_{i,j \in S} x_{ij}^v \leq |S| - 1 \quad S \subseteq N \setminus \{0\}; 2 \leq |S| \leq n-1; v = 1, \dots, NV \quad (2.7)$$

$$y_i^v \in \{0,1\} \quad i = 1, \dots, n ; v = 1, \dots, NV \quad (2.8)$$

$$x_{ij}^v \in \{0,1\} \quad i = 1, \dots, n ; j = 1, \dots, n ; v = 1, \dots, NV \quad (2.9)$$

O objetivo do modelo (2.1) é a minimização dos custos (ou distâncias) totais de viagem, satisfazendo todas as restrições impostas pelo problema.

A restrição (2.2) representa a capacidade dos veículos, ou seja, limita a utilização do veículo à sua capacidade máxima ( $Q_v$ ).

A restrição (2.3) garante que todos os veículos iniciem e terminem suas rotas no depósito ( $i = 0$ ). A restrição (2.4) garante que todos os clientes serão atendidos e que cada um deles será visitado por apenas um veículo.

As restrições (2.5) e (2.6) são as restrições de conservação dos fluxos, ou seja, cada cliente terá obrigatoriamente um único arco de entrada e um único arco de saída correspondente. Esta restrição garante também que os veículos não interrompam as suas rotas em um cliente.

A restrição (2.7) impede a formação de subrotas (*subtours*), impondo para cada veículo, que os arcos de qualquer subgrafo  $S$  do grafo  $G(N,A)$  que compõe o problema, não formem circuitos isolados. A forte natureza combinatória envolvida na formação de todos os subgrafos de um problema reforça a complexidade e dificuldade computacional na resolução de problemas de roteirização de veículos (VIANNA, 2002).

As restrições (2.8) e (2.9) garantem que as variáveis  $x_{ij}^v$  e  $y_i^v$  sejam binárias.

### 2.1.2. Problema de roteirização de veículos com entregas fracionadas (SDVRP)

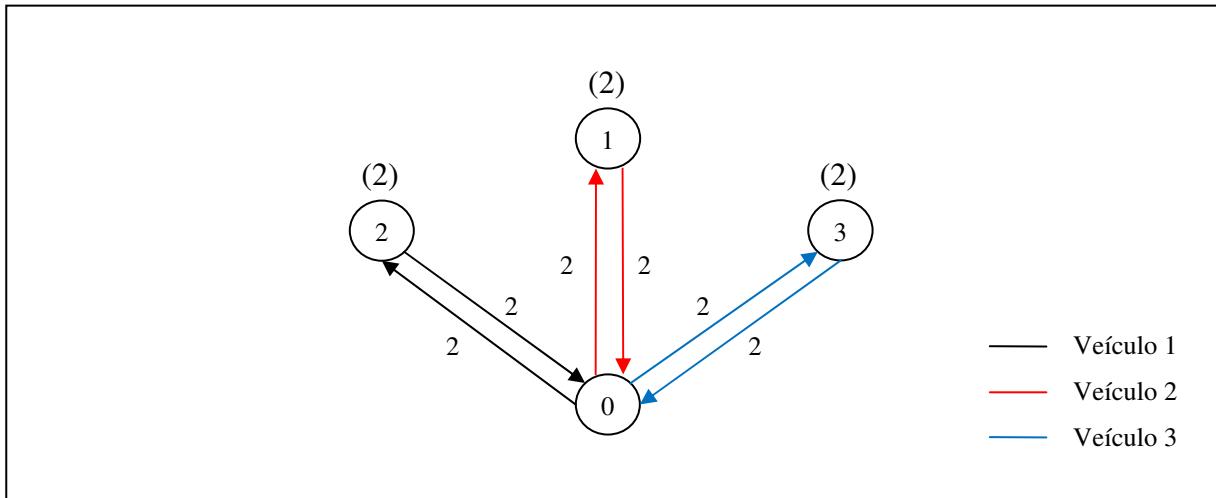
O problema de roteirização de veículos com entregas fracionadas (SDVRP – *Split Delivery Vehicle Routing*) é uma relaxação do problema clássico de roteirização, no qual uma

entrega para um determinado ponto de demanda pode ser realizada por mais de um veículo, podendo a demanda em cada ponto ser maior ou menor que a capacidade dos veículos (DROR; TRUDEAU, 1990).

Logicamente, a necessidade de se utilizar mais de um veículo nos casos em que a demanda de um ponto excede a capacidade máxima dos veículos é obvia. No entanto, mesmo em situações em que as demandas de todos os clientes são menores ou iguais à demanda dos veículos, a utilização de mais de um veículo para atender um único cliente pode trazer benefícios (ARCHETTI *et al.*, 2001).

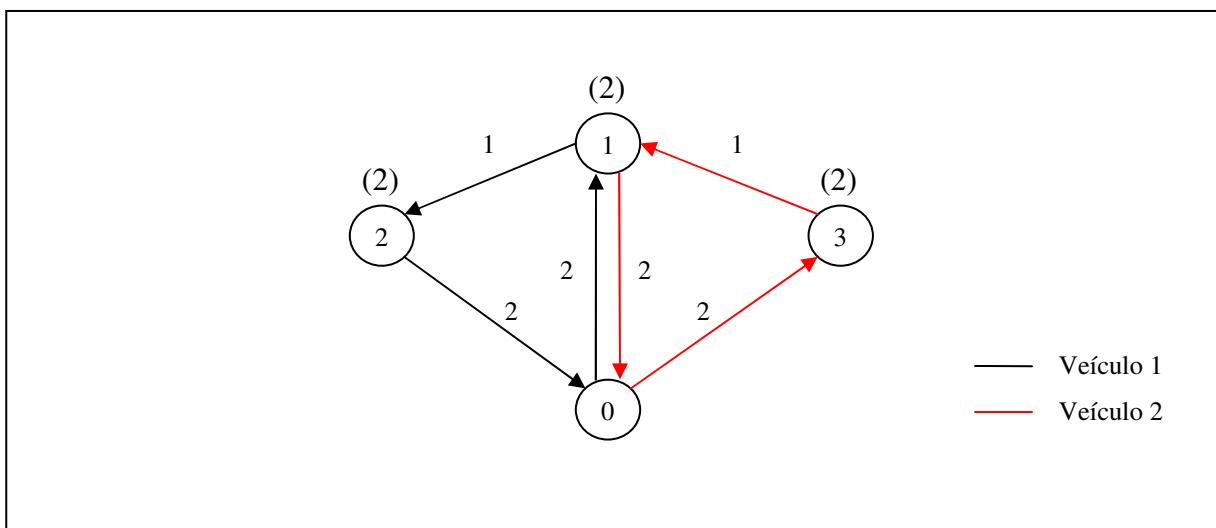
Segundo Dror e Trudeau (1990), para problemas com demandas muito pequenas (até 10% da capacidade dos veículos) não há diferenças significativas entre as soluções obtidas nas duas versões (clássica e entregas fracionadas). Contudo, no estudo realizado, nos conjuntos de problemas onde as demandas de cada cliente eram maiores que 10% da capacidade dos veículos, obtiveram-se significativas reduções de custo, no caso das entregas fracionadas. Essas economias se traduziram tanto em termos de distância total percorrida quanto no número de veículos utilizados.

O exemplo mostrado nas figuras 2.1.1 e 2.2.2 a seguir, extraído de Chen *et al.* (2006), apresenta uma comparação entre os dois modelos de roteirização de veículos (clássico e entregas fracionadas), ilustrando as principais diferenças entre os dois problemas, bem como as economias obtidas com a relaxação da restrição de visitas únicas para cada cliente. No grafo que representa o exemplo a seguir, há três clientes (representados pelos nós 1,2 e 3) de demandas idênticas, de duas unidades cada um (indicadas pelos números em parênteses acima de cada cliente) que devem ser atendidos por veículos partindo de um depósito central, indicado pelo nó 0 (zero). Os custos (ou distâncias) de viagem são representados pelos números ao lado de cada arco e os veículos possuem capacidade igual a três unidades.



*Figura 2.1.1 – Solução de um problema de roteirização de veículos sem entregas fracionadas – Extraído de Chen et al. (2006)*

A situação apresentada na figura 2.1.1, mostra o problema resolvido através da abordagem clássica de roteirização de veículos, no qual cada cliente deve ser atendido por apenas um veículo. Observa-se que neste caso, a solução ótima do problema é composta por três rotas, onde cada uma delas é realizada por um veículo diferente. Vale ressaltar que cada veículo é capaz de atender apenas um cliente, uma vez que possuem capacidade de três unidades e a demanda de cada cliente é de duas unidades. O custo total de viagem é de 12 unidades (monetárias ou de distância).



*Figura 2.1.2 – Solução de um problema de roteirização de veículos com entregas fracionadas– Extraído de Chen et al. (2006)*

A situação da figura 2.1.2 mostra a solução do problema de entregas fracionadas, no qual a demanda de cada cliente individual pode ser atendida por mais de um veículo. Nota-se que no caso das entregas fracionadas, a solução ótima tem custo total de 10 unidades e é composta por apenas duas rotas, em contraste com as três rotas da solução do problema clássico de roteirização. Pode-se observar que neste caso a demanda do cliente 1 foi dividida entre os dois veículos (uma unidade para cada veículo), possibilitando a utilização da capacidade máxima dos veículos (3 unidades).

Essa diferença se traduz na obtenção de economias, tanto no número de veículos utilizados (são necessários apenas dois veículos para efetuar as entregas contra três no problema clássico), quanto no custo total de viagem (10 unidades contra 12 unidades no problema clássico), além de um melhor aproveitamento da capacidade dos veículos.

### **2.1.3. Complexidade computacional do SDVRP**

Os problemas de roteirização de veículos, por possuírem um caráter fortemente combinatório, apresentam grande dificuldade de resolução através de métodos exatos. De fato, a grande maioria dos problemas de roteirização de veículos se enquadra na categoria de complexidade *NP-Hard* (NP-Difícil em português), inclusive o problema clássico de roteirização e o problema do caixeiro viajante (LENSTRA; RINNOOY KAN, 1981). Isto significa que o tempo de resolução e processamento dos problemas enquadrados nessa categoria cresce exponencialmente de acordo com o seu tamanho. Dessa forma, recorre-se freqüentemente à utilização de métodos heurísticos para a resolução de problemas de otimização combinatória, tais como os problemas de roteirização de veículos, viabilizando em termos práticos a obtenção de soluções próximas da ótima, tanto com relação ao tempo de processamento, quanto à utilização de recursos computacionais.

#### **A classe de problemas *NP-Hard***

De maneira informal, a classe de complexidade computacional *NP-Hard* é definida como: “classe de problemas ao menos tão difíceis quanto o problema mais difícil na classe NP”. Os problemas da classe NP são ditos como solucionáveis em tempo polinomial não-determinístico. Dessa forma, embora seja provável que não exista solução para um problema *NP-Hard* em tempo polinomial, ainda não foi provado que de fato todos os problemas no

universo *NP-Hard* não possam ser resolvidos em tempo polinomial (GAREY; JOHNSON, 1979).

Devido ao fato de serem uma relaxação do problema tradicional de roteirização de veículos, os problemas de roteirização com entregas fracionadas apresentam uma maior dificuldade na obtenção de soluções exatas, uma vez que a possibilidade de fracionamento de entregas potencializa o caráter combinatório do problema.

Dror e Trudeau (1990) mostram que o SDVRP também se enquadra na categoria *NP-Hard*. De fato, um caso particular do SDVRP, no qual a capacidade do menor veículo é maior que a soma de todas as demandas e cuja matriz  $\{C_{ij}\}$  dos custos de viagem satisfaz a desigualdade triangular (cuja definição é a seguinte: dados três pontos  $i, j$  e  $k$ , a distância  $d_{ij}$  entre os pontos  $i$  e  $j$ , os mais afastados entre si é menor, ou no máximo igual à soma das distâncias dos outros pontos,  $d_{ik}, d_{jk}$ ), é exatamente igual ao problema do caixeiro viajante, conhecido por ser da classe *NP-Hard*. Em outras palavras, qualquer instância de um problema do caixeiro viajante pode ser transformada nessa instância do SDVRP e resolvida como tal. Portanto, por redução (ver GAREY; JOHNSON, 1979), prova-se que o SDVRP é da classe *NP-Hard*.

## 2.2. FORMULAÇÃO MATEMÁTICA DO SDVRP

A formulação matemática do problema de roteirização com entregas fracionadas foi introduzida na literatura por Dror e Trudeau (1990) na publicação “*Split Delivery Routing*”. Esta formulação apresenta características fundamentais bastante semelhantes à formulação de Fisher e Jaikumar (1981), visto que o SDVRP é uma derivação direta do problema clássico de roteirização.

Esta seção dedica-se a apresentar a formulação do SDVRP, tal como mostrada por Dror e Trudeau (1990) e Dror, Laporte e Trudeau (1994). Primeiramente, serão apresentados os parâmetros, as variáveis, a função objetivo e as restrições do problema e em seguida, cada componente da formulação será explicado detalhadamente.

### 2.2.1. Parâmetros e formulação do SDVRP

A seguir, é apresentada a formulação completa do problema de roteirização com entregas fracionadas (SDVRP).

### Parâmetros

- $NV$  = Número de veículos  
 $n$  = Número de clientes para os quais uma entrega deve ser feita. Os clientes são indexados de 1 a  $n$  e o índice 0 representa o depósito central.  
 $Q_v$  = Capacidade do veículo  $v$   
 $q_i$  = Demanda do cliente  $i$   
 $c_{ij}$  = Custo de viagem do cliente  $i$  para o cliente  $j$   
 $S$  = Representa um subgrafo qualquer do problema, excluindo o depósito

### Variáveis

$y_i^v$  = Fração de demanda do cliente  $i$  entregue pelo veículo  $v$ . Representa a parcela da demanda total de um determinado cliente  $i$  que será entregue por um veículo  $v$ .

$$x_{ij}^v = \begin{cases} 1, & \text{se o veículo } v \text{ viaja diretamente do cliente } i \text{ para o cliente } j \\ 0, & \text{caso contrário} \end{cases}$$

### Formulação

$$\min \sum_{i=0}^n \sum_{j=0}^n \sum_{v=1}^{NV} c_{ij} x_{ij}^v \quad (2.10)$$

s.a.

$$\sum_{i=0}^n x_{ip}^v - \sum_{j=0}^n x_{pj}^v = 0 \quad p = 0, \dots, n ; v = 1, \dots, NV \quad (2.11)$$

$$\sum_{v=1}^{NV} y_i^v = 1 \quad i = 1, \dots, n \quad (2.12)$$

$$\sum_{i=1}^n q_i y_i^v \leq Q_v \quad v = 1, \dots, NV \quad (2.13)$$

$$\sum_{j=0}^n x_{ij}^v \geq y_i^v \quad i = 1, \dots, n ; v = 1, \dots, NV \quad (2.14)$$

$$\sum_{v=1}^{NV} \sum_{i, j \in S} x_{ij}^v \leq |S| - 1 \quad S \subseteq N \setminus \{0\}; |S| \geq 2 \quad (2.15)$$

$$0 \leq y_i^v \leq 1 \quad i = 1, \dots, n ; v = 1, \dots, NV \quad (2.16)$$

$$x_{ij}^v \in \{0,1\} \quad i = 1, \dots, n ; j = 1, \dots, n ; v = 1, \dots, NV \quad (2.17)$$

### 2.2.2. Detalhamento do modelo

Este item tem com objetivo apresentar detalhadamente cada componente da formulação do SDVRP, ou seja, a função objetivo e as restrições do problema. As explicações virão acompanhadas de exemplos ilustrativos, visando facilitar a compreensão, mostrando a aplicação de cada restrição ao contexto do problema.

$$\text{Função Objetivo (2.10): } \min \sum_{i=0}^n \sum_{j=0}^n \sum_{v=1}^{NV} c_{ij} x_{ij}^v$$

O objetivo do modelo é a minimização dos custos (ou distâncias) totais de transporte, atendendo a todas as restrições impostas pelo problema. Cada arco do grafo  $G(N,A)$  que compõe o problema, possui um custo de viagem ( $c_{ij}$ ) associado, correspondente ao deslocamento do nó  $i$  para o nó  $j$ . Neste modelo, admite-se, salvo contrário, que os custos de viagem são iguais para todo veículo  $v$  e que a matriz de custos (ou distâncias) é simétrica, isto é  $c_{ij} = c_{ji}$ . Dessa forma, para todo arco  $x_{ij}^v$  que faz parte da solução do problema (ou seja, a variável  $x_{ij}^v$  recebe o valor 1), haverá um custo de viagem  $c_{ij}$  associado a este arco. Para os demais arcos (que não fazem parte da solução), a variável  $x_{ij}^v$  recebe o valor zero e consequentemente o custo de viagem associado será nulo. O custo total é a soma de todos os custos associados aos arcos que fazem parte da solução, ou seja, todos os arcos onde  $x_{ij}^v = 1$ .

$$\text{Restrição (2.11): } \sum_{i=0}^n x_{ip}^v - \sum_{j=0}^n x_{pj}^v = 0; \quad p = 0, \dots, n; \quad v = 1, \dots, NV$$

A restrição (2.11) impõe a conservação dos fluxos de entrada e saída, ou seja, se um veículo  $v$  é designado para atender um cliente  $p$ , partindo diretamente do cliente  $i$ , isto é  $x_{ip}^v = 1$ , este veículo deve obrigatoriamente sair do cliente  $p$  em direção a um destino  $j$  (seja outro cliente ou o próprio depósito), forçando a variável  $x_{pj}^v$  a assumir o valor 1. Caso contrário, se o cliente  $p$  não for designado a fazer parte da solução (rota) do veículo  $v$ , os arcos receberão valor nulo. Esta restrição, portanto, impede que os veículos interrompam suas rotas em um cliente. A figura 2.2.1 a seguir ilustra esta situação.

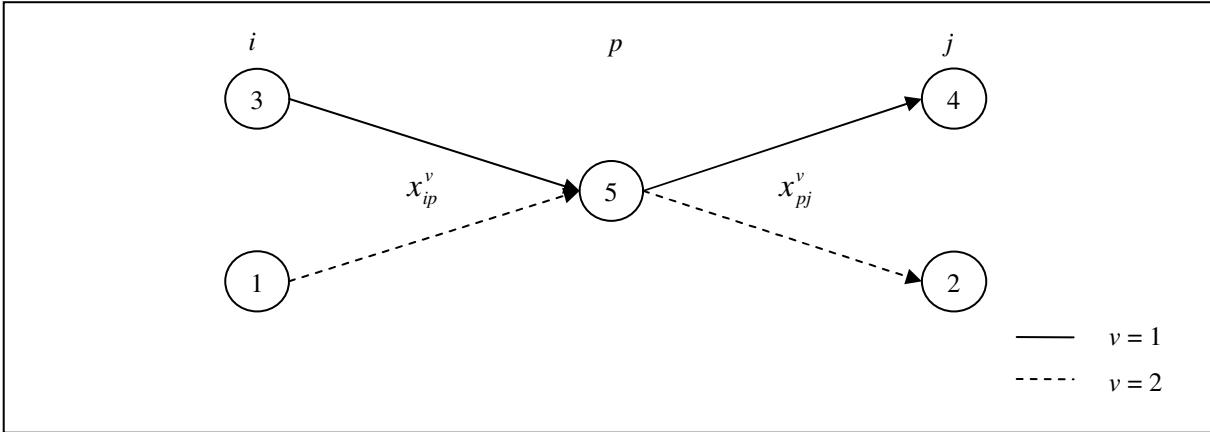


Figura 2.2.1 – Conservação dos fluxos de entrada e saída

Observa-se na figura 2.2.1, que para o veículo 1 (representado pela linha contínua), o nó 5 tem um arco de entrada ( $x_{3,5}^1$ , ou seja, o arco que sai do nó 3 em direção ao nó 5) e um arco de saída ( $x_{5,4}^1$ ). O mesmo se observa para o veículo 2 (linha tracejada).

Cabe ressaltar que a restrição (2.11), apesar de garantir que não haja interrupção das rotas dos veículos em um cliente, permite que mais de um veículo atenda determinado cliente, diferentemente do que ocorre no modelo de Fisher e Jaikumar (1981), conforme as restrições (2.5) e (2.6). Este par de restrições impõe que para todo cliente haverá apenas um único arco de entrada e um único arco de saída. A restrição (2.5) garante que se um determinado cliente  $j$  for atendido por um veículo  $v$  (isto é,  $y_j^v = 1$ ), haverá um único arco entrando ( $\sum_i x_{ij}^v = 1$ ) neste cliente, ou seja, todo cliente  $j$  será atendido apenas uma vez por um único veículo. A restrição (2.6) garante que se um determinado cliente  $i$  for atendido por um veículo  $v$  (isto é,  $y_i^v = 1$ ), haverá apenas um arco saindo deste cliente ( $\sum_j x_{ij}^v = 1$ ).

Dessa forma, a principal diferença entre o problema clássico de roteirização e o SDVRP reside na possibilidade de fracionar entregas para um cliente (ou seja, um cliente pode ser atendido por mais de um veículo) e pode ser observada na comparação entre a restrição (2.11) e o par de restrições (2.5) e (2.6).

**Restrição (2.12):**  $\sum_{v=1}^{NV} y_i^v = 1; i = 1, \dots, n$

A restrição (2.12) garante que toda a demanda de cada um dos clientes será atendida, ou seja, a soma das frações da demanda de um cliente  $i$  ( $y_i^v$ ) entregues por todos os veículos  $v$

deve totalizar a demanda total ( $q_i$ ) deste cliente, isto é,  $q_i \sum_{v=1}^{NV} y_i^v = q_i$ , ou  $\sum_{v=1}^{NV} y_i^v = 1$ . Neste

modelo, a variável  $y_i^v$  pode assumir qualquer valor no intervalo [0;1], sendo que este valor representa a parcela (fração) da demanda do cliente  $i$  entregue pelo veículo  $v$ . O exemplo da figura 2.2.2 a seguir mostra a demanda do cliente 5 (as demandas dos clientes estão indicadas pelos números nos parênteses) fracionada entre os dois veículos (idênticos e de capacidade igual a 3 unidades).

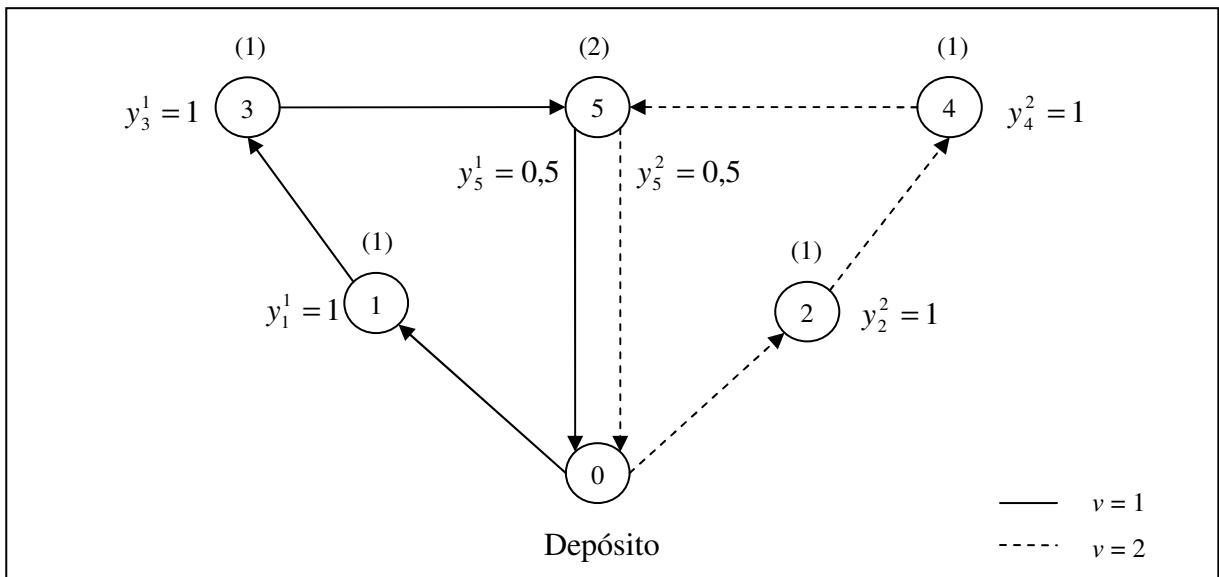


Figura 2.2.2 – Todas as demandas dos clientes serão atendidas, sendo permitidas entregas fracionadas

Nota-se que na formulação de Fisher e Jaikumar (1981) a variável  $y_i^v$  é binária, impondo que cada entrega realizada para um cliente  $i$  pelo veículo  $v$  seja completa, de tal maneira que apenas um veículo é forçado a atender toda a demanda de um determinado cliente. Portanto, a formulação do problema clássico de roteirização não admite que a demanda individual de um dado cliente exceda a capacidade dos veículos, o que não ocorre no problema de roteirização com entregas fracionadas.

$$\text{Restrição (2.13): } \sum_{i=1}^n q_i y_i^v \leq Q_v ; v = 1, \dots, NV$$

A restrição (2.13) impede que as entregas designadas a um determinado veículo  $v$  excedam sua capacidade máxima, ou seja, a soma de todas as entregas designadas a um veículo  $v$ , obtida através da soma das frações de demanda do cliente  $i$  a ser entregue pelo veículo  $v$  ( $y_i^v$ ), multiplicadas por suas respectivas demandas ( $q_i$ ), não deve ultrapassar a capacidade máxima do veículo  $v$  ( $Q_v$ ). Do exemplo mostrado na figura 2.2.2, observa-se que:

$$q_1 y_1^1 + q_3 y_3^1 + q_5 y_5^1 \leq Q_1$$

$$q_2 y_2^2 + q_4 y_4^2 + q_5 y_5^2 \leq Q_2$$

$$\text{Restrição (2.14): } \sum_{j=0}^n x_{ij}^v \geq y_i^v ; i = 1, \dots, n ; v = 1, \dots, NV$$

A restrição (2.14) impõe que, se um determinado ponto de demanda for atendido por um veículo, então necessariamente o mesmo veículo deve sair deste ponto.

Esta restrição pode ser interpretada da seguinte forma: se um cliente  $i$  qualquer for atendido por um veículo  $v$  (isto é  $y_i^v > 0$ ), então obrigatoriamente haverá um arco  $x_{ij}^v$ , saindo do cliente  $i$  em direção a um cliente  $j$ , ou seja, o veículo  $v$  sai do cliente  $i$  para o destino  $j$ . Por outro lado, se o cliente  $i$  não for atendido pelo veículo  $v$  ( $y_i^v = 0$ ), então não haverá nenhum fluxo saindo do cliente  $i$ . A figura 2.2.3 a seguir ilustra tal situação.

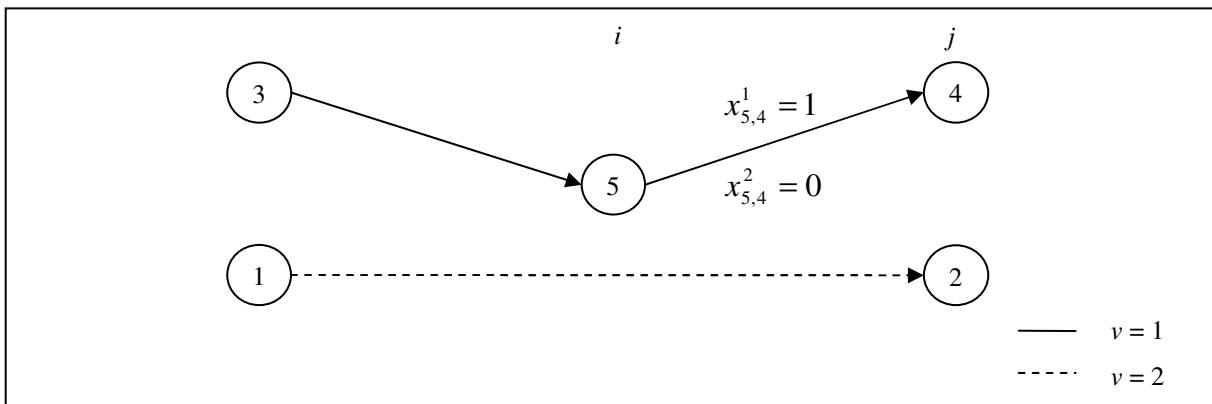


Figura 2.2.3 – Se um cliente  $i$  é atendido por um veículo  $v$ , haverá obrigatoriamente um arco que sai dele

Nota-se que aplicando o somatório de todos os veículos ( $v=1, \dots, NV$ ) à restrição (2.14) e combinando com a restrição (2.12), obtém-se a seguinte relação:

$$\sum_{v=1}^{NV} \sum_{j=0}^n x_{ij}^v \geq \sum_{v=1}^{NV} y_i^v = 1 \quad i = 1, \dots, n \quad (2.18)$$

Esta restrição garante que cada cliente receberá pelo menos uma visita de algum veículo, ou seja, para cada cliente  $i$ , existirá pelo menos um arco  $x_{ij}^v$ , que sai da origem  $i$  para o destino  $j$ , implicando que no mínimo um veículo  $v$  passa obrigatoriamente pelo cliente  $i$ . A restrição (2.18) impede que haja uma configuração como a mostrada na figura 2.2.4.

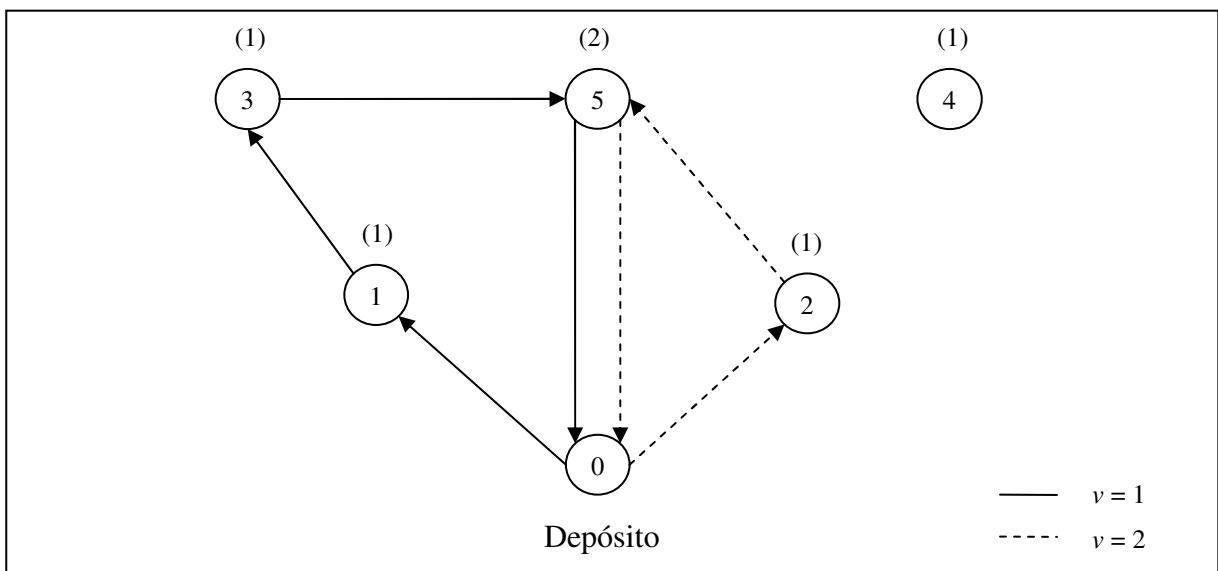


Figura 2.2.4 – Configuração não permitida pela restrição (2.18)

**Restrição (2.15):**  $\sum_{v=1}^{NV} \sum_{i,j \in S} x_{ij}^v \leq |S| - 1; S \subseteq N \setminus \{0\}; |S| \geq 2$

A restrição (2.15) garante que não haja a formação de subrotas (*subtours*), ou circuitos isolados (que não passam pelo depósito central) em qualquer subgrafo  $S \subseteq N \setminus \{0\}$ , de  $G(N,A)$ , o grafo principal do problema. A figura 2.2.5 ilustra a formação de um circuito fechado em um subgrafo de  $G$ .

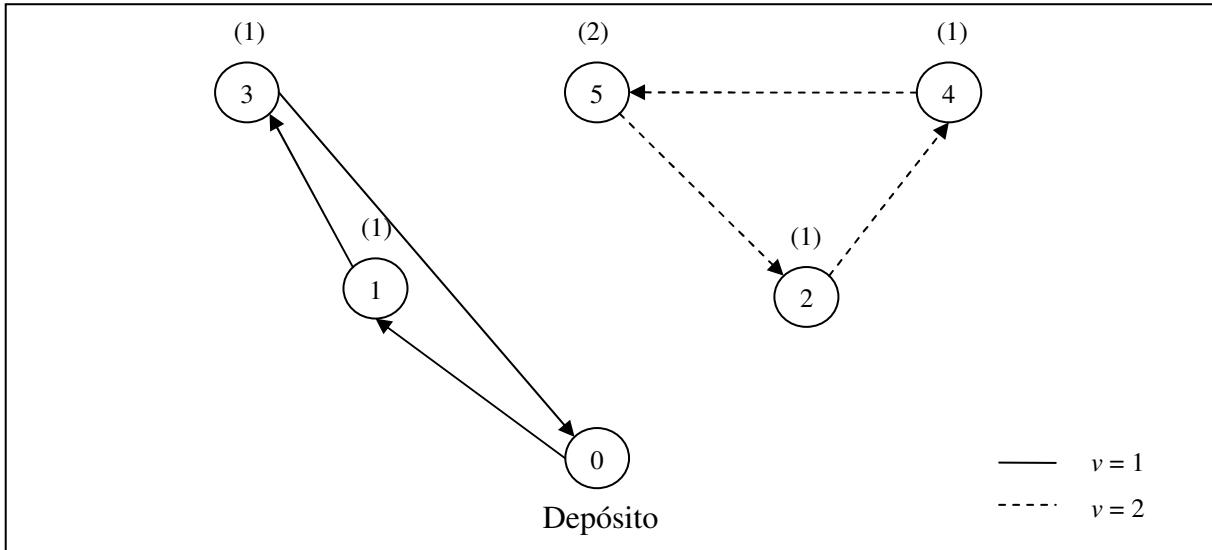


Figura 2.2.5 – Formação de uma subrota no grafo \$G(N,A)\$

A seguir será apresentada a derivação da restrição (2.15) a partir da restrição de eliminação de subrotas do problema clássico de roteirização de veículos, conforme trabalho de Dror, Laporte e Trudeau (1994).

Na formulação de Fisher e Jaikumar (1981) do problema clássico de roteirização, a restrição de eliminação de subrotas é dada pela relação (2.7).

$$\sum_{i,j \in S} x_{ij}^v \leq |S| - 1 \quad S \subseteq N \setminus \{0\}; 2 \leq |S| \leq n-1; v = 1, \dots, NV \quad (2.7)$$

De acordo com Dror, Laporte e Trudeau (1994), a restrição (2.7) é equivalente à restrição (2.19) a seguir.

$$\sum_{v=1}^{NV} \sum_{i,j \in S} x_{ij}^v \leq |S| - V(S) \quad S \subseteq N \setminus \{0\}; |S| \geq 2; v = 1, \dots, NV \quad (2.19)$$

Na restrição (2.19), \$V(S)\$ representa o número mínimo de veículos necessário para atender todos os nós de um dado subgrafo \$S\$ do problema, excluindo o depósito. Esta restrição elimina dois tipos de configurações não permitidas: (i) subrotas não ligadas ao depósito central e (ii) rotas ligadas ao depósito, mas cuja demanda excede a capacidade dos veículos. No entanto, observa-se que a restrição (2.19) não se aplica ao SDVRP. O exemplo a seguir ilustrado na figura 2.2.6 (extraído de DROR; LAPORTE; TRUDEAU, 1994) mostra que a restrição (2.19) não é satisfeita.

Seja  $S = \{1, 2, 3, 4, 5\}$ ,  $Q_v = 3$  para todo  $v$ ,  $q_1 = q_2 = q_3 = q_4 = 1$  e  $q_5 = 2$ . Observa-se que o número de veículos necessário para satisfazer a demanda se  $S$  é  $V(S) = 2$ . A configuração da figura 2.2.6 mostra uma solução possível do problema (dividindo-se a demanda do cliente 5 entre os dois veículos). Porém, aplicando a restrição (2.19) ao subgrafo  $S$ , obtém-se:

$$\begin{aligned} & x_{1,3}^1 + x_{1,3}^2 + x_{3,5}^1 + x_{3,5}^2 + x_{5,4}^1 + x_{5,4}^2 + x_{4,2}^1 + x_{4,2}^2 + x_{2,1}^1 + x_{2,1}^2 + \dots \\ & \dots + x_{2,4}^1 + x_{2,4}^2 + x_{4,5}^1 + x_{4,5}^2 + x_{5,3}^1 + x_{5,3}^2 + x_{3,1}^1 + x_{3,1}^2 + x_{1,2}^1 + x_{1,2}^2 \leq |S| - V(S) \end{aligned}$$

Da solução possível do problema, tem-se:  $x_{1,3}^1 = x_{3,5}^1 = x_{4,5}^2 = x_{2,4}^2 = 1$ , e ainda:  $|S| = 5$  e  $V(S) = 2$ . Dessa forma, obtém-se:  $x_{1,3}^1 + x_{3,5}^1 + x_{4,5}^2 + x_{2,4}^2 \leq 5 - 2$ , ou seja,  $4 \leq 3$ , mostrando que a restrição (2.19), não é válida para o SDVRP.

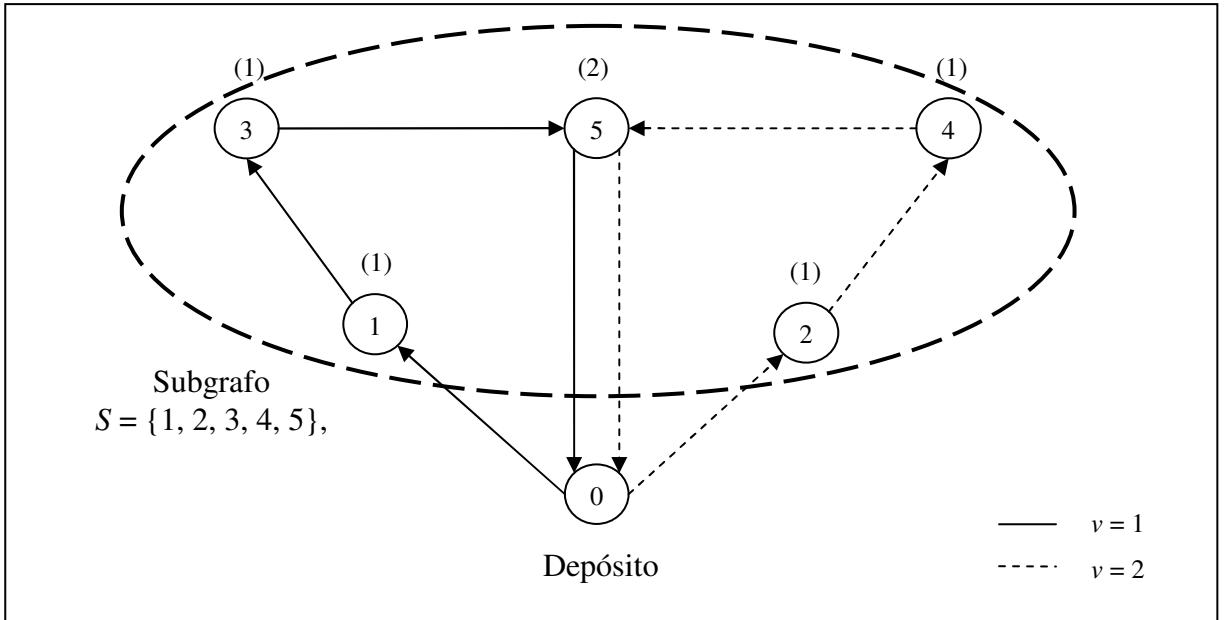


Figura 2.2.6 –Contra-exemplo mostrando que a restrição (2.19) não é válida para o SDVRP

Dror, Laporte e Trudeau (1994) mostram então que a seguinte equivalência à restrição (2.19) é válida para o SDVRP.

$$\sum_{v=1}^{NV} \sum_{i,j \in S} x_{ij}^v \leq \sum_{i \in S} d_i - V(S) \quad S \subseteq N \setminus \{0\}; |S| \geq 2 \quad (2.20)$$

Na restrição (2.20),  $d_i$  pode ser interpretado como o grau do vértice  $i$ , ou seja, o número de arcos que saem do nó  $i$  para o nó  $j$  e é dado por:

$$d_i = \sum_{v=1}^{NV} \sum_{j=0}^n x_{ij}^v \quad (i \in N) \quad (2.21)$$

A restrição (2.20) também elimina os dois tipos de configuração apresentados na restrição (2.19). A eliminação da configuração (i), ou seja, subrotas não ligadas ao depósito central é particularmente importante, uma vez que na formulação apresentada no item 2.2.1, não há nenhuma outra restrição que força a saída e o retorno de um veículo ao depósito, diferentemente do que ocorre na formulação de Fisher e Jaikumar (1981), com a restrição (2.3).

Finalmente, Dror, Laporte e Trudeau (1994), mostram que, se a matriz dos custos de viagem  $C=\{c_{ij}\}$  satisfaz a desigualdade triangular, então a seguinte inequação é uma restrição de eliminação de subrotas válida para o SDVRP.

$$\sum_{v=1}^{NV} \sum_{i,j \in S} x_{ij}^v \leq |S| - 1 \quad S \subseteq N \setminus \{0\}; |S| \geq 2 \quad (2.15)$$

Nota-se que a restrição (2.15) é bastante semelhante à restrição (2.7) de eliminação de subrotas da formulação de Fisher e Jaikumar (1981), com a diferença de que em (2.15) para cada subgrafo  $S$  considera-se os arcos de todos os veículos e não os arcos de cada veículo separadamente.

A restrição (2.15) tem como vantagem uma maior facilidade de modelagem, sem a necessidade de se calcular parâmetros adicionais, como ocorre com a restrição (2.20). Dessa forma, ela é particularmente importante no contexto do presente trabalho, uma vez que todos os problemas de roteirização encontrados na empresa satisfazem a desigualdade triangular, possibilitando a sua aplicação e consequentemente simplificando o processo de modelagem.

### 3. MÉTODO DE RESOLUÇÃO

Os problemas de roterização de veículos se caracterizam pela grande complexidade de resolução, em virtude de uma forte natureza combinatória, o que os enquadra na classe de problemas *NP-Hard*. Dessa forma, quanto maior o número de pontos considerados, o tempo de resolução computacional dos problemas cresce exponencialmente, o que inviabiliza a aplicação de métodos exatos de resolução em problemas de maiores dimensões. Para efeitos de esclarecimento, o número de restrições de eliminação de subrotas em um exemplo com 20 pontos de demanda é da ordem de  $O(2^n)$ , ou seja, mais de 1.000.000 restrições (GOLDBARG; LUNA, 2000), o que equivale à todas as combinações possíveis de subrotas de 2 a 19 pontos.

Neste contexto, os métodos e algoritmos heurísticos surgem como uma maneira viável de resolver tais problemas, garantindo soluções próximas à ótima e com tempos de computação razoáveis e bastante viáveis do ponto de vista prático. Em uma empresa como a TRADBRAS, onde, apesar dos problemas de roteirização serem relativamente pequenos (raramente os problemas ultrapassam 30 pontos de entrega), a dinâmica de entregas de mercadorias se dá em uma base diária e a necessidade de um modelo ágil, que forneça boas soluções em um tempo reduzido torna-se ainda mais evidente e importante.

Assim sendo, o modelo de resolução selecionado para o presente trabalho é um método heurístico bastante eficiente para a resolução de problemas de roterização de veículos e que possui uma grande flexibilidade para adaptação ao caso particular do problema considerado no trabalho, ou seja, o problema de roteirização de veículos com entregas fracionadas (SDVRP). Trata-se do algoritmo de varredura (*sweep algorithm*), introduzido por Gillett e Miller (1974), que envolve uma lógica relativamente simples, mas bastante eficiente na resolução de problemas de roteirização.

A estrutura deste capítulo encontra-se dividida em três partes: na primeira, será apresentada, para fins ilustrativos, uma abordagem iterativa no processo de eliminação de subrotas, que possibilita a obtenção de soluções ótimas para o SDVRP em tempos razoáveis para problemas com número pequeno de pontos (até 15 pontos). A segunda parte é dedicada à apresentação detalhada da heurística de varredura de Gillett e Miller, com cada etapa de construção do algoritmo e a adaptação necessária para aplicação no SDVRP. Finalmente, na terceira parte será apresentado um exemplo reduzido, extraído de uma situação real da empresa, como forma de ilustrar a aplicação do algoritmo da varredura, bem como a aplicação

do procedimento iterativo apresentado na primeira parte para a obtenção da solução ótima. As duas soluções (exata e heurística) serão então comparadas.

### **3.1. MÉTODO EXATO PARA RESOLUÇÃO DO SDVRP**

Conforme mencionado anteriormente, a resolução de problemas de roteirização através de métodos exatos está condicionada à forte natureza combinatória que reside nas restrições de formação de subrotas, o que inviabiliza em muitos casos a obtenção de soluções ótimas para o problema. No entanto, em termos práticos, muitas dessas restrições apresentam uma folga considerável no ponto de ótimo e não serão violadas na solução ótima do problema.

Neste sentido, Pataki (2003), apresenta um procedimento iterativo para resolução do problema do caixeiro viajante que se baseia no fato de que nem todas as restrições de eliminação de subrotas devem ser incluídas na formulação do problema. Isto se deve ao fato de que, apesar do grande número de restrições de eliminação de subrotas, apenas algumas delas estão efetivamente ativas na solução ótima. De acordo com este modelo, as restrições de eliminação de subrotas vão sendo adicionadas à medida que ocorre uma violação (isto é, a formação de uma subrota) na solução do problema de roteirização.

Vale ressaltar que este procedimento foi originalmente aplicado a um problema de caixeiro viajante, mas também pode ser aplicado a problemas semelhantes, como por exemplo, os problemas de roteirização de veículos (PATAKI, 2003).

#### **Adaptação da formulação do SDVRP**

Como mostrado no capítulo anterior, a formulação de Dror, Laporte e Trudeau (1994) do SDVRP não contém nenhuma restrição que explicitamente force a saída e o retorno dos veículos ao depósito. A restrição (2.15) apresentada anteriormente, implicitamente impede a formação de subrotas desconexas do depósito central. Deste modo, para que a solução do problema tenha obrigatoriamente as rotas dos veículos ligadas ao depósito central, deve-se verificar todas, ou pelo menos grande parte das restrições de eliminação de subrotas. Assim sendo, torna-se necessária uma adaptação da formulação original do SDVRP de modo que o procedimento iterativo possa ser aplicado.

Esta adaptação é bastante simples e envolve a introdução de duas restrições adicionais que obrigam a saída e o retorno dos veículos ao depósito central.

*Restrição de saída do depósito*

$$\sum_{j=1}^n x_{0j}^v = 1 \quad v = 1, \dots, NV \quad (3.1)$$

A restrição (3.1) garante que o veículo  $v$  sairá do depósito, ou seja, haverá um arco  $x_{0j}^v$  saindo do nó 0 (zero) em direção a um cliente  $j$ .

*Restrição de retorno ao depósito*

$$\sum_{i=1}^n x_{i0}^v = 1 \quad v = 1, \dots, NV \quad (3.2)$$

A restrição (3.2) garante que o veículo  $v$  retorne ao depósito, ou seja, haverá um arco  $x_{i0}^v$  retornando do cliente  $i$  ao depósito (nó 0).

**Aplicação do procedimento iterativo**

O exemplo mostrado na figura 3.1.1 a seguir ilustra a aplicação deste procedimento iterativo em um problema de roteirização de veículos com entregas fracionadas. O problema do exemplo tem 8 pontos de demanda, identificados pelos pontos azuis. O depósito é representado pelo ponto vermelho, localizado no centro. Supondo que, resolvendo o problema de roteirização através da formulação de Dror, Laporte e Trudeau, sem utilizar as restrições de eliminação de subrotas, a solução ótima do problema apresente a configuração apresentada na figura 3.1.1.

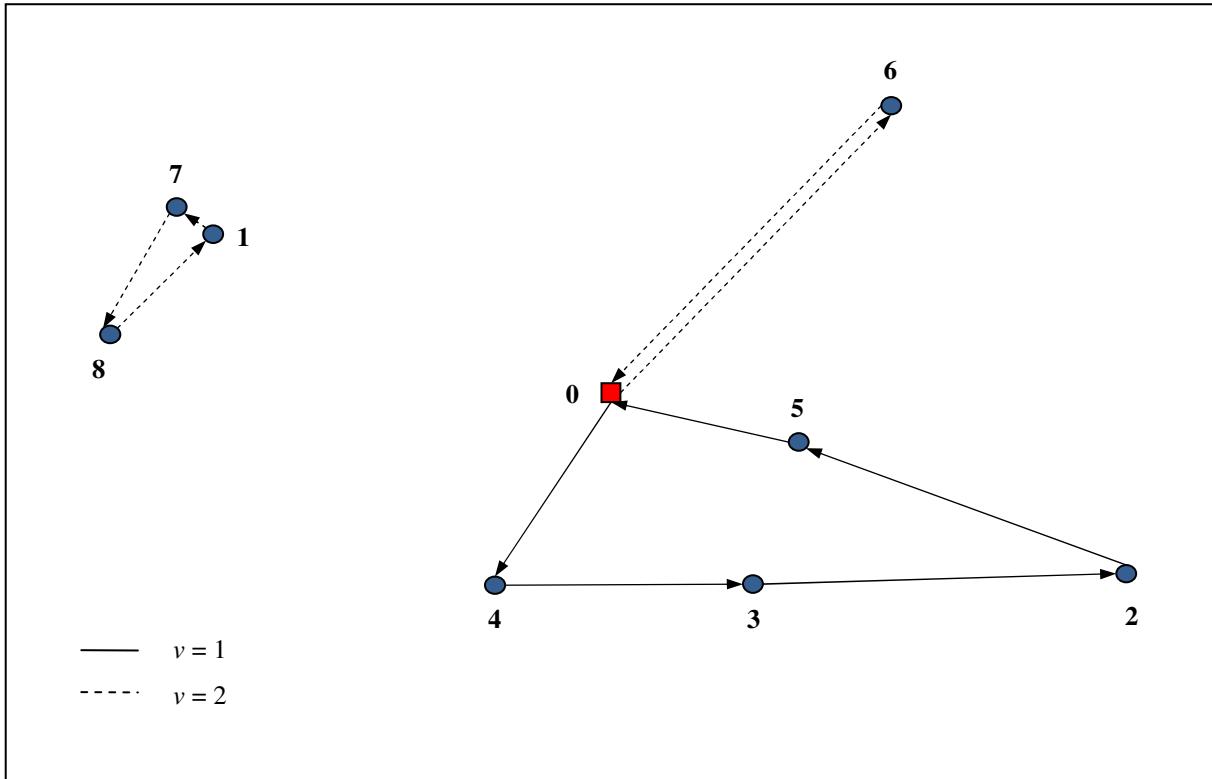


Figura 3.1.1 – Exemplo de aplicação do processo iterativo de eliminação de subrotas – primeira iteração

Na figura, as linhas contínuas representam a rota do veículo 1 e as linhas tracejadas representam a rota do veículo 2. Pode-se observar que houve a formação de uma subrota na solução designada ao veículo 2, formada pelos pontos 1, 7 e 8. Esta subrota pode ser eliminada aplicando-se as seguintes restrições:

$$x_{1,7}^2 + x_{7,8}^2 + x_{8,1}^2 \leq 2$$

$$x_{1,8}^2 + x_{8,7}^2 + x_{7,1}^2 \leq 2$$

A segunda restrição é necessária para eliminar também a subrota no sentido inverso, que em geral tem o mesmo custo (quando os custos de viagem entre dois pontos quaisquer são iguais em ambos os sentidos, isto é  $c_{ij} = c_{ji}$ ). Após a aplicação desta restrição, resolve-se o problema novamente, obtendo-se a solução mostrada na figura 3.1.2.

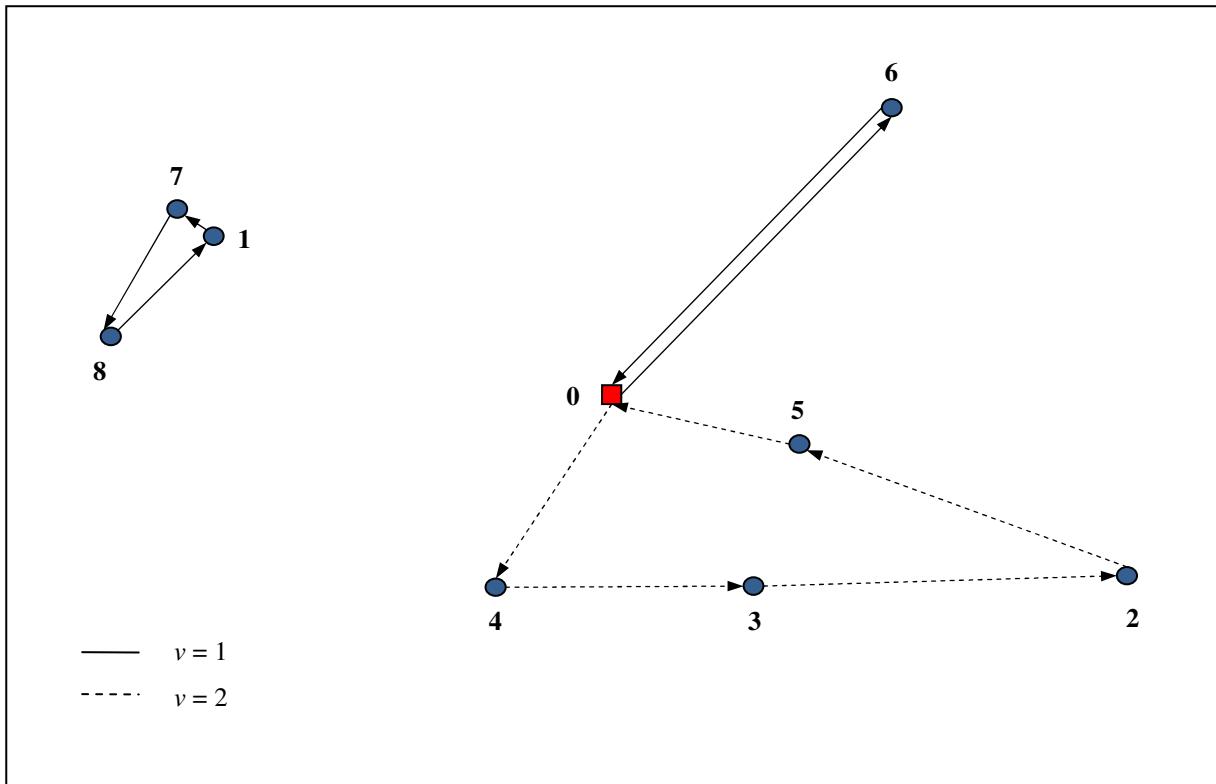


Figura 3.1.2 – Exemplo de aplicação do processo iterativo de eliminação de subrotas – segunda iteração

Pode-se observar que nesta solução o que ocorreu foi uma troca na designação das rotas, ou seja, a rota que na solução anterior foi designada para o veículo 1 passou a ser a rota do veículo 2 e vice-versa. Dessa forma as restrições aplicadas no passo anterior também devem ser aplicadas para o veículo 1, como segue:

$$x_{1,7}^1 + x_{7,8}^1 + x_{8,1}^1 \leq 2$$

$$x_{1,8}^1 + x_{8,7}^1 + x_{7,1}^1 \leq 2$$

Resolvendo o problema novamente, obtém-se a solução da figura 3.1.3.

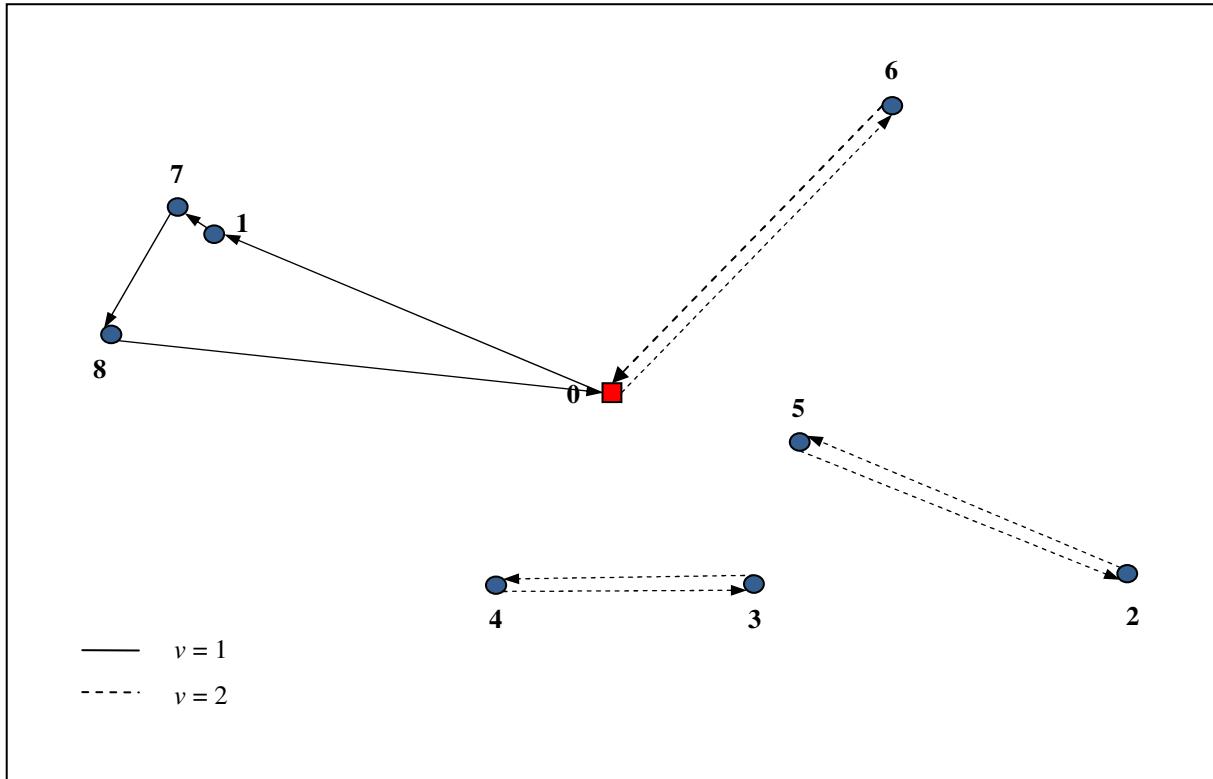


Figura 3.1.3 – Exemplo de aplicação do processo iterativo de eliminação de subrotas – terceira iteração

Nesta solução há duas subrotas: 3-4-3 e 3-5-2, neste ponto, pode-se optar por eliminar apenas uma delas ou as duas simultaneamente. Escolhendo eliminar apenas a subrota 3-4-3, tem-se:

$$x_{3,4}^1 + x_{4,3}^1 \leq 2$$

$$x_{3,4}^2 + x_{4,3}^2 \leq 2$$

A solução do problema fornece a configuração da figura 3.1.4.

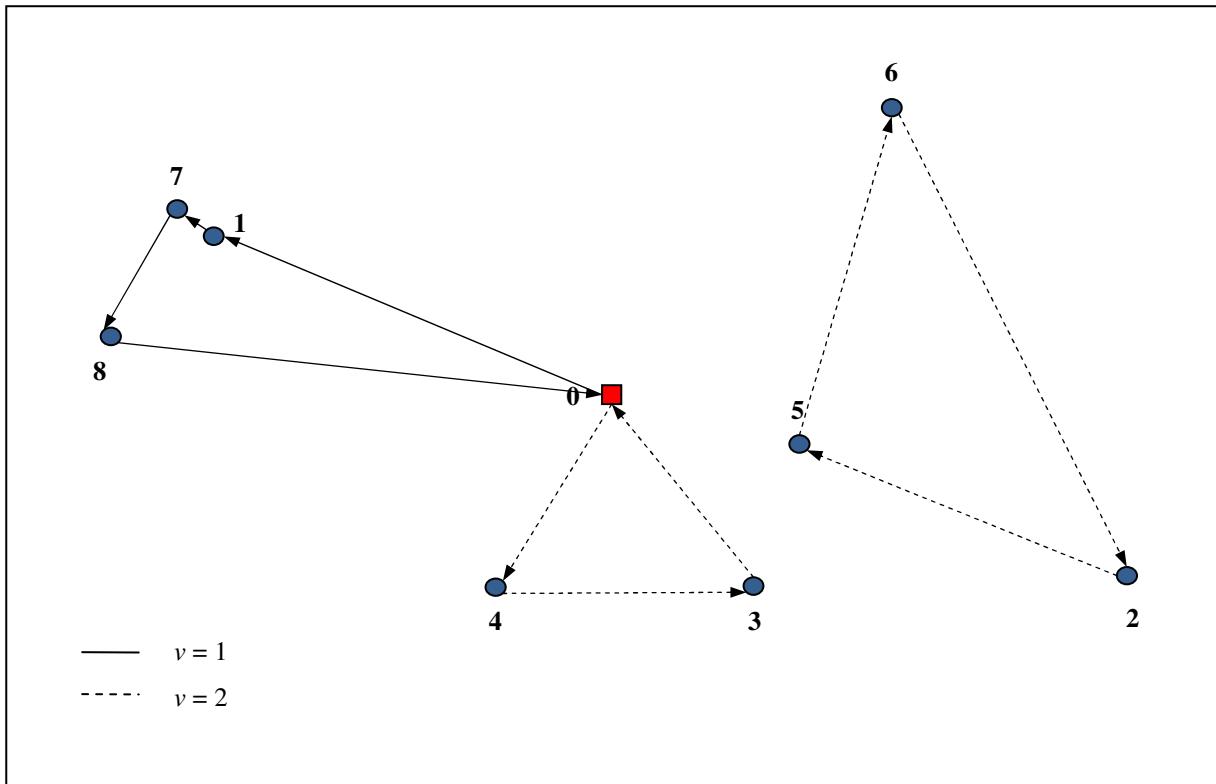


Figura 3.1.4 – Exemplo de aplicação do processo iterativo de eliminação de subrotas – quarta iteração

Nesta solução houve a formação da subrota 5-6-2-5, no percurso do veículo 2. Aplicando as seguintes restrições de eliminação para os dois veículos:

$$x_{5,6}^1 + x_{6,2}^1 + x_{2,5}^1 \leq 2$$

$$x_{5,6}^2 + x_{6,2}^2 + x_{2,5}^2 \leq 2$$

$$x_{5,2}^1 + x_{2,6}^1 + x_{6,5}^1 \leq 2$$

$$x_{5,2}^2 + x_{2,6}^2 + x_{6,5}^2 \leq 2$$

Obtém-se a solução ótima do problema conforme a figura 3.1.5:

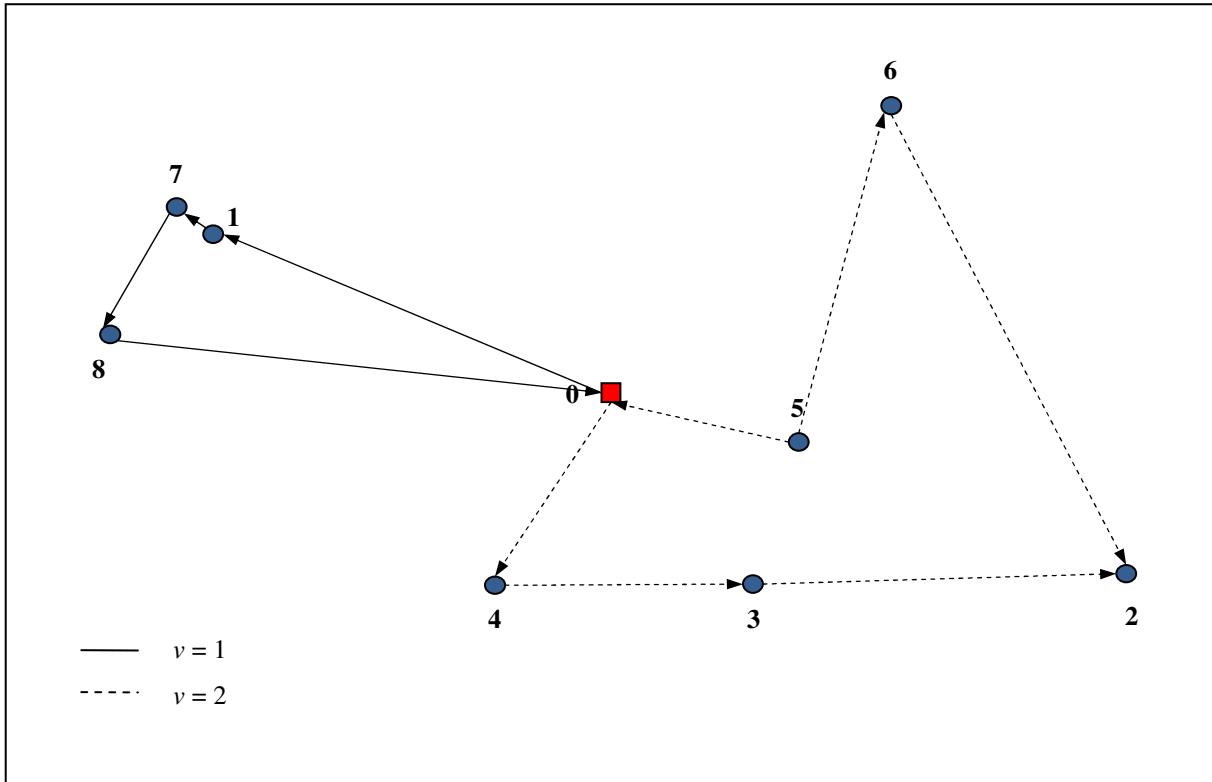


Figura 3.1.5 – Exemplo de aplicação do processo iterativo de eliminação de subrotas – solução ótima

Nota-se neste exemplo ilustrativo que para obter a solução ótima do problema, sem a formação de subrotas, foi necessária a inclusão de apenas 8 restrições adicionais de eliminação, ao passo que um problema com essas dimensões (8 pontos de demanda mais o depósito central) possui um número total de restrições de eliminação de subrotas da ordem de 250. Vale ressaltar que a quantidade necessária de restrições de eliminação de subrotas para a obtenção da solução ótima varia conforme o tamanho do problema e a localização (disposição) espacial dos pontos no problema considerado.

De acordo com os experimentos realizados no decorrer do presente trabalho, há uma tendência das subrotas se formarem em pontos que estão próximos entre si e distantes dos demais pontos, conforme pode ser observado no exemplo anterior no caso da subrota formada pelos pontos 1, 7 e 8. Além disso, os experimentos realizados indicam que, apesar de ser um método bastante eficiente para a obtenção da solução ótima em problemas de roteirização de veículos, o tempo de computação cresce drasticamente, em problemas maiores que 15 pontos de demanda. Isto se deve ao aumento exponencial no número de variáveis inteiras do problema, com o aumento do número de pontos. Em termos práticos, isto significa que os

problemas vão se tornando intratáveis computacionalmente à medida que o tamanho do problema aumenta.

Sendo assim, do ponto de vista da dinâmica das operações de entrega e das necessidades da empresa, uma solução ágil, porém acurada e flexível é mais adequada, como é o caso do método heurístico que será descrito na seção seguinte. No entanto, o processo iterativo apresentado mostra-se como uma ferramenta bastante importante na obtenção de um parâmetro de comparação para um método heurístico eficiente, de modo a verificar se uma solução obtida heuristicamente é realmente próxima do ótimo (PATAKI, 2003).

### **3.2. O ALGORITMO DE VARREDURA**

O algoritmo de varredura (*sweep algorithm*) foi introduzido na literatura por Gillett e Miller (1974) na publicação “*A Heuristic Algorithm for the Vehicle Dispatching Problem*”. Trata-se de um método que segue uma abordagem do tipo duas-fases (*cluster-first, route-second*, ou agrupa-roteiriza), no qual os pontos de demanda são inicialmente subdivididos em grupos (*clusters*) e dentro de cada grupo, roteiros individuais são determinados em cada um dos grupos previamente determinados (BELFIORE, 2006).

O modelo de resolução proposto para o desenvolvimento do presente trabalho foi selecionado por atender uma série de necessidades e requisitos do processo de entrega de mercadorias da empresa estudada, além de ser bastante eficiente, obtendo soluções muito boas para problemas pequenos e médios, de até 30 pontos (GILLETT; MILLER, 1974). O algoritmo tem a vantagem de possuir uma lógica relativamente simples, o que facilita bastante sua implantação e, embora tenha sido originalmente desenvolvido para o problema clássico de roteirização de veículos, tem uma boa flexibilidade de adaptação para a variação de entregas fracionadas abordada neste trabalho.

O algoritmo de varredura foi um dos primeiros métodos heurísticos de resolução de problemas de roteirização de veículos a utilizar a abordagem de subdividir os pontos do problema em grupos menores, desdobrando o problema principal em uma série de subproblemas menores (*cluster-first, route-second*). Isto se traduz em menores tempos de computação, o que é de bastante utilidade para os interesses da empresa, uma vez que a rotina diária de entregas exige um modelo ágil e ao mesmo tempo eficiente. Além disso, esta característica do modelo faz com que sua aplicabilidade seja bastante simples para uma série

de plataformas, como a do *software Microsoft Excel*, bastante difundida e utilizada em todo o mundo, além de ser facilmente acessível para qualquer empresa.

No entanto, este modelo também apresenta algumas desvantagens, exatamente devido à sua natureza de dois estágios, pois não considera diretamente questões de tempo, como o tempo total de ciclo e a incorporação de restrições de janelas de tempo. Estas desvantagens, porém, não têm relevância para a empresa estudada, uma vez que a principal restrição das operações de entrega na empresa diz respeito à capacidade dos veículos e não ao tempo de viagem. Além disso, a grande maioria dos clientes não trabalha com tempos de início e fim de atendimento pré-determinados (janelas de tempo).

Assim colocado, observa-se que o algoritmo de varredura é um modelo bastante adequado para a resolução do problema abordado no trabalho e a sua razoável facilidade de implantação torna este modelo bastante interessante para as necessidades da empresa estudada. A tabela 3.2.1 a seguir, compila algumas das principais características com respeito à resolução de problemas de roteirização de veículos atendidas ou não pelo algoritmo de varredura e também mostra seu alinhamento às necessidades da empresa estudada.

*Tabela 3.2.1 – Comparação entre o método de varredura e a heurística de Clarke & Wright*

Parâmetro	Algoritmo de varredura	Necessidade da empresa
<b>Tempo de computação</b>	Baixos para problemas de pequeno e médio porte	Necessita de soluções ágeis, pois as entregas ocorrem diariamente
<b>Proximidade da solução em relação ao ótimo</b>	Boa para problemas de pequeno e médio porte (LAPORTE <i>et al.</i> , 2000)	Exige-se apenas que as soluções obtidas tragam economias consideráveis em relação à situação atual
<b>Facilidade de implantação e utilização</b>	Modelo de fácil implantação e utilização	Necessária, pois não há na empresa pessoal especializado para o planejamento dos roteiros
<b>Flexibilidade do modelo</b>		
<b>Restrições de tempo de viagem</b>	Não considera diretamente	Normalmente as entregas terminam dentro das jornadas de trabalho
<b>Restrições de janelas de tempo</b>	Não considera diretamente	Os clientes atendidos não trabalham com janelas de tempo
<b>Flexibilidade para o SDVRP</b>	Fácil adaptação	Necessária, pois é a proposta do presente trabalho

A partir de uma análise da tabela 3.2.1 pode-se concluir que, apesar de não tratar diretamente alguns aspectos como a incorporação de restrições adicionais de janelas de tempo, o algoritmo de varredura é bastante adequado para as necessidades da empresa. Principalmente devido ao fato de se tratar de um modelo de fácil implementação e que possui a flexibilidade necessária para adaptação ao problema de roteirização de entregas fracionadas estudado no presente trabalho, além de fornecer boas soluções em tempos de computação bastante reduzidos (LAPORTE *et al.*, 2000).

### 3.2.1. Funcionamento do algoritmo de varredura

O algoritmo de varredura de Gillett e Miller em sua forma geral é construído em duas fases principais: a primeira consiste na formação de grupos de pontos a partir da rotação de um raio centrado no depósito. A segunda fase consiste na determinação de um roteiro a ser percorrido por cada veículo designado a atender um dos grupos determinados na etapa anterior, através da resolução de um problema de caixeiro viajante.

O algoritmo completo possui duas partes principais, a varredura para frente (*forward sweep*), que executa a varredura a partir da origem no sentido anti-horário e a varredura para trás (*backward sweep*), que executa a varredura no sentido inverso (horário). A lógica de funcionamento do algoritmo de varredura será apresentada mais detalhadamente a seguir.

Para auxiliar o entendimento do algoritmo, a seguinte notação, adaptada de Gillett e Miller (1974), será utilizada:

- $n$  = Número de pontos do problema. Os clientes são indexados de 1 a  $n$  e o índice 0 representa o depósito central.
- $q_i$  = Demanda do cliente  $i$
- $(x_i, y_i)$ : Coordenadas retangulares do ponto  $i$ , onde  $i = 0, \dots, n$
- $Q$  = Capacidade dos veículos (o algoritmo original considera veículos de mesma capacidade)
- $d_{ij}$  = Distância entre os pontos  $i$  e  $j$ , onde  $i = 0, \dots, n$ ;  $j = 0, \dots, n$
- $\theta_i$  = Ângulo de coordenada polar do ponto  $i$ , onde  $i = 0, \dots, n$
- $r_i$  = Raio a partir do depósito (ponto 0) ao ponto  $i$ , onde  $i = 0, \dots, n$

O ângulo de coordenada polar é calculado através da equação (3.3):

$$\theta_i = \operatorname{arctg} \left( \frac{y_i - y_0}{x_i - x_0} \right) \quad (3.3)$$

Onde:  $-\pi < \theta_i < 0$  se  $y_i - y_0 < 0$ , e  $0 \leq \theta_i \leq \pi$ , se  $y_i - y_0 \geq 0$ .

As etapas do método da varredura para frente (*forward sweep*) são descritas a seguir.

**Passo 1:** Classificar os pontos representados por suas coordenadas polares  $(\theta_i, r_i)$  em ordem crescente, em relação ao eixo  $x$  de coordenadas cartesianas, de modo que  $\theta_i < \theta_{i+1}$ . Se houver algum  $i$  e  $j$  tal que  $\theta_i < \theta_j$ , então  $i < j$ , se  $d_{0-i} < d_{0-j}$ .

**Passo 2 (inicialização de rota):** Selecionar um veículo vazio  $v$ , ainda não utilizado.

**Passo 3 (construção de rota):** Partindo do último ponto designado a fazer parte da rota atual, rotacionar o ângulo  $\theta = \theta_i$ , para  $\theta = \theta_{i+1}$ , enquanto a soma das demandas dos pontos atendidos não exceder a capacidade do veículo  $v$ . Se ainda houver pontos não atendidos, voltar ao passo 2.

**Passo 4 (otimização de rota):** Otimizar separadamente cada rota determinada no passo anterior, através da resolução do correspondente problema de caixeiro viajante (através de métodos exatos ou aproximados). O custo total da solução do problema de roteirização corresponde à soma dos custos de cada solução dos problemas de caixeiro viajante determinados no passo 3.

**Passo 5:** Rotacionar o eixo  $(x,y)$  no sentido anti-horário de tal maneira que o primeiro ponto se torna o último, o segundo ponto se torna o primeiro, o terceiro se torna o segundo e assim por diante. Voltar ao passo 2, até que todas as configurações de eixos tenham sido testadas, ou seja, todos os pontos tenham sido tomados como ponto de partida. A solução do algoritmo de varredura para frente corresponde à solução de menor custo encontrada entre todas as configurações de eixos. Uma adaptação a ser feita neste passo do modelo é reduzir a configuração dos eixos a apenas 4 possibilidades, aplicando uma rotação de  $90^\circ$  em relação à configuração anterior.

As figuras 3.2.1, 3.2.2, 3.2.3 e 3.2.4 a seguir mostram a mecânica de funcionamento do algoritmo de varredura.

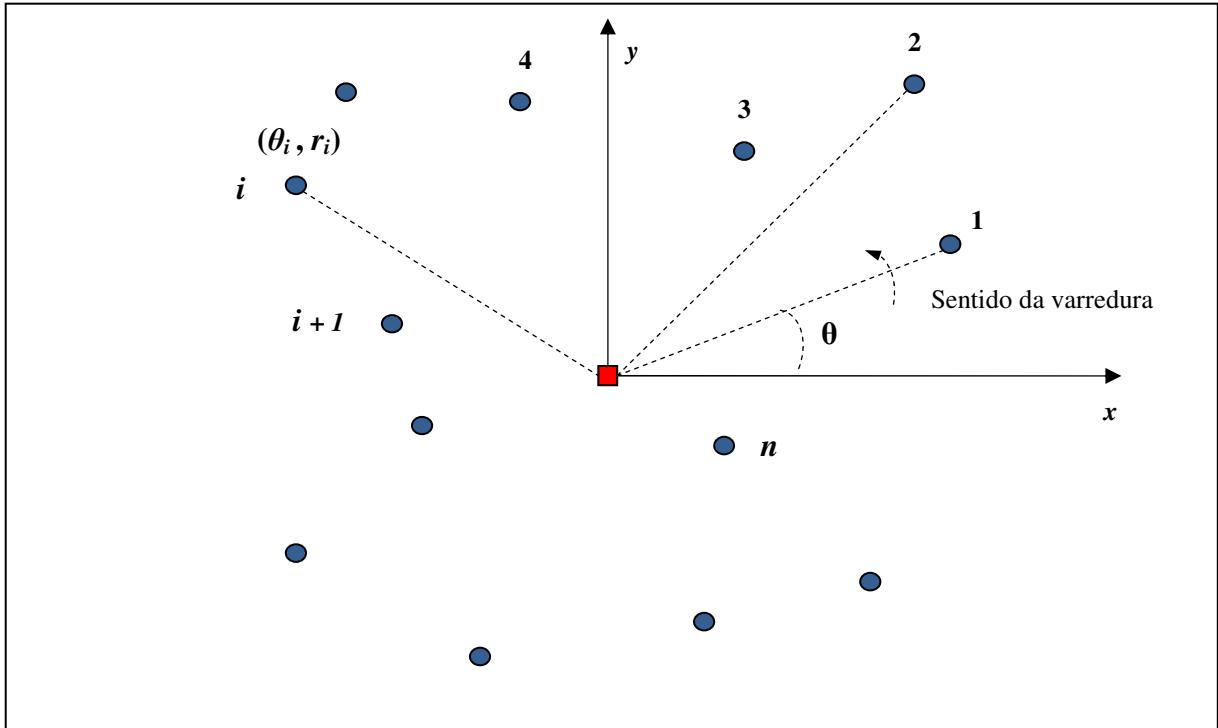


Figura 3.2.1 – Classificação dos ângulos de coordenadas polares em ordem crescente

A figura 3.2.1 mostra o passo 1 do método de varredura. Como se pode observar, os pontos são classificados em ordem crescente de acordo com seus respectivos ângulos em coordenadas polares  $(\theta, r)$ . A varredura é então realizada conforme o sentido apresentado na figura (anti-horário), do menor ângulo para o maior. O critério de parada da varredura de um determinado veículo é a sua capacidade, ou seja, a varredura é executada até que não haja mais capacidade ociosa no veículo e se houver pontos ainda não atendidos, a varredura continua com um novo veículo vazio.

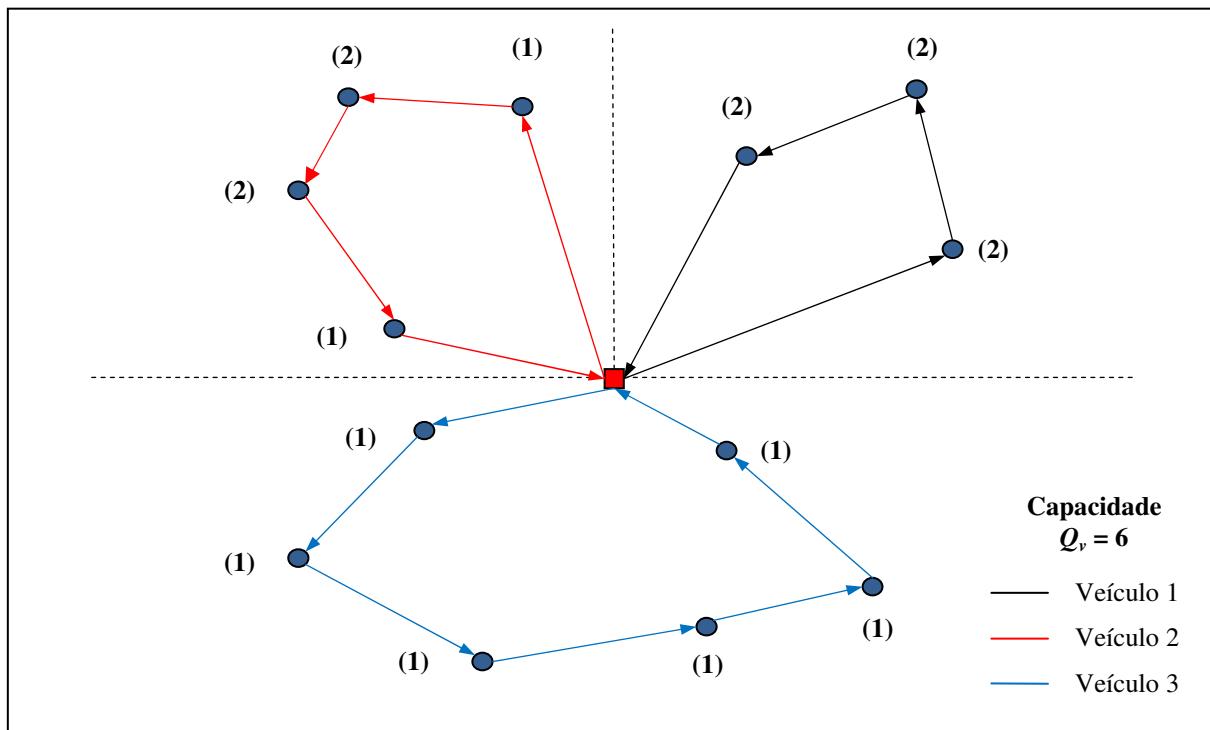


Figura 3.2.2 – Roteiros formados com o método da varredura

A figura 3.2.2 mostra os roteiros formados após a varredura de todos os pontos. Dentro de cada grupo de pontos (*clusters*) determinados pela varredura, o correspondente problema de caixeiro viajante é resolvido (com métodos exatos ou aproximados), de modo a obter a configuração apresentada na figura. O custo total do problema de roteirização é a soma dos custos de cada subroteiro determinado pela varredura.

Neste trabalho, os problemas de caixeiro viajante serão resolvidos através de um método exato, utilizando a formulação de Millet, Tucker e Zemlin (1960), que é apresentada no Apêndice A.

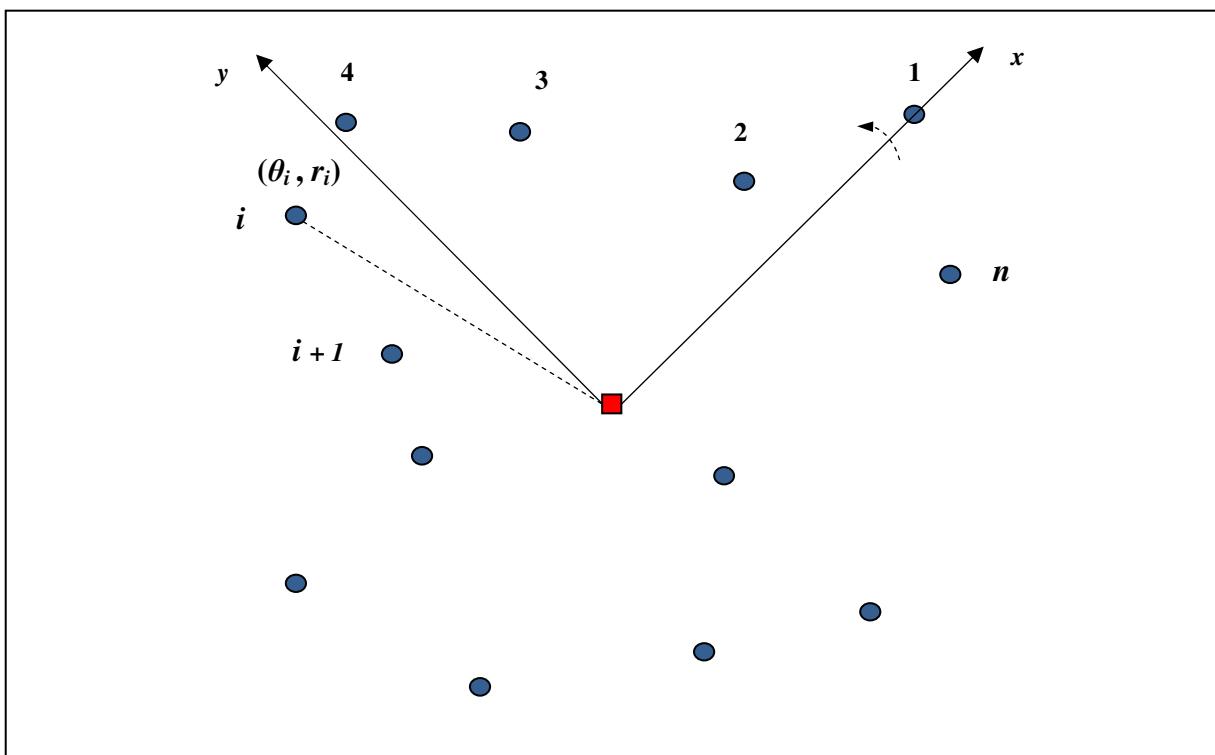


Figura 3.2.3 – Rotação do eixo ( $x,y$ ) no sentido anti-horário

A figura 3.2.3 mostra o passo 5 do método da varredura. Neste passo, o eixo ( $x,y$ ) é rotacionado no sentido-anti horário, de tal maneira que o primeiro ponto passe a ser o último, o segundo passe a ser o primeiro e assim por diante. Após essa reclassificação dos pontos, a varredura é executada novamente e roteiros diferentes são formados. Este procedimento é repetido até que todas as configurações possíveis de eixos sejam testadas. A solução de menor custo entre das configurações de eixos testadas será a escolhida.

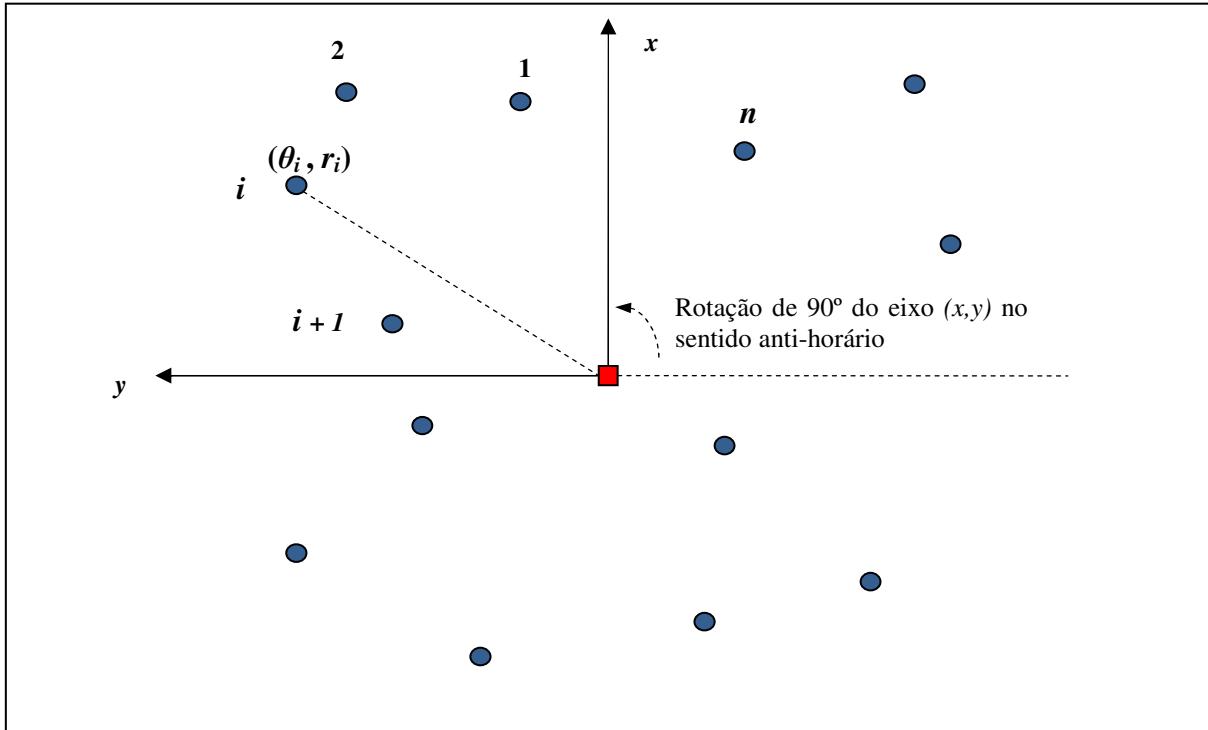


Figura 3.2.4 – Adaptação do passo de rotação do eixo ( $x,y$ )

A figura 3.2.4, apresenta ilustrativamente a adaptação do passo 5 a ser utilizada neste trabalho. Optou-se por realizar esta adaptação para reduzir o número de iterações do modelo, visando obter uma redução nos tempos computacionais e também de maneira a facilitar a modelagem do algoritmo. Nesta adaptação, ao invés de rotacionar o eixo ( $x,y$ ) no sentido anti-horário até o próximo cliente, o eixo será rotacionado em 90° no sentido anti-horário em relação à última posição, conforme mostrado na figura. Com essa adaptação, o número de configurações possíveis será reduzido a quatro configurações.

Opcionalmente, pode-se realizar a segunda parte do algoritmo, a varredura para trás (*backward sweep*), que é similar à varredura para frente, com exceção do fato de que os roteiros são formados na ordem inversa, ou seja, do último ponto classificado em termos de coordenadas polares (o ponto de maior ângulo), no sentido horário.

### 3.2.2. Adaptação do método da varredura ao SDVRP

A adaptação do algoritmo da varredura ao problema de roteirização de veículos com entregas fracionadas é bastante simples e envolve poucas alterações no algoritmo original. Basicamente, esta adaptação consiste em permitir a divisão da demanda de um cliente, caso a demanda do cliente  $i$ , definido como o cliente seguinte na ordem da varredura de um veículo  $v$

seja maior que a capacidade restante deste veículo. Esta modificação permite que o veículo  $v$  utilize sua capacidade máxima, atendendo parte da demanda do cliente  $i$  e deixando o restante da demanda deste cliente para o próximo veículo.

Na prática, a adaptação ocorre da seguinte maneira:

Se, durante a varredura dos pontos da rota de um determinado veículo  $v$ , o próximo ponto  $i$  a ser percorrido pela varredura tiver demanda ( $d_i$ ) maior que a capacidade restante  $Q_r$  do veículo (isto é, se o veículo  $v$  não tiver capacidade suficiente para atender a demanda do cliente  $i$ ), então o ponto  $i$  será dividido em dois pontos imaginários ( $i'$  e  $i''$ ), na mesma posição do ponto  $i$ , sendo que um deles possui demanda igual à capacidade restante ( $Q_r$ ) do veículo  $v$  e o outro possui a demanda restante do ponto  $i$  original, ou seja,  $d_i - Q_r$ .

Esta adaptação possibilita uma melhor utilização da capacidade dos veículos, bem como permite o fracionamento de entregas, conforme a formulação de Dror, Laporte e Trudeau (1994), adequando o modelo original às características do problema abordado no trabalho.

As figuras 3.2.5 e 3.2.6 a seguir ilustram as diferenças entre o algoritmo de varredura original e a adaptação proposta para o SDVRP.

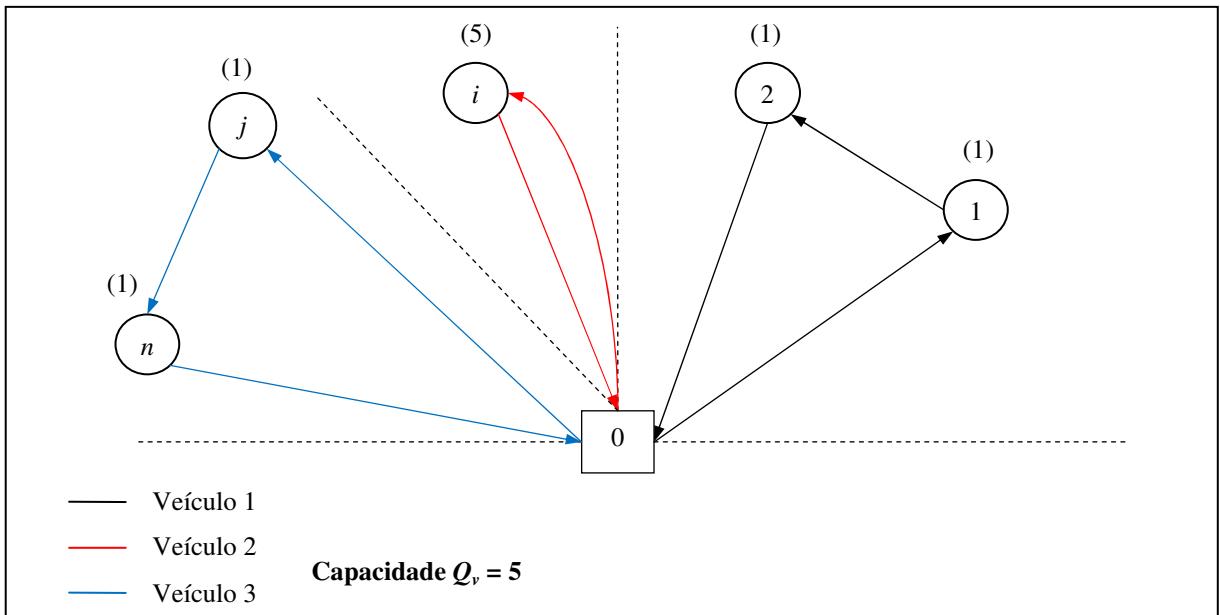
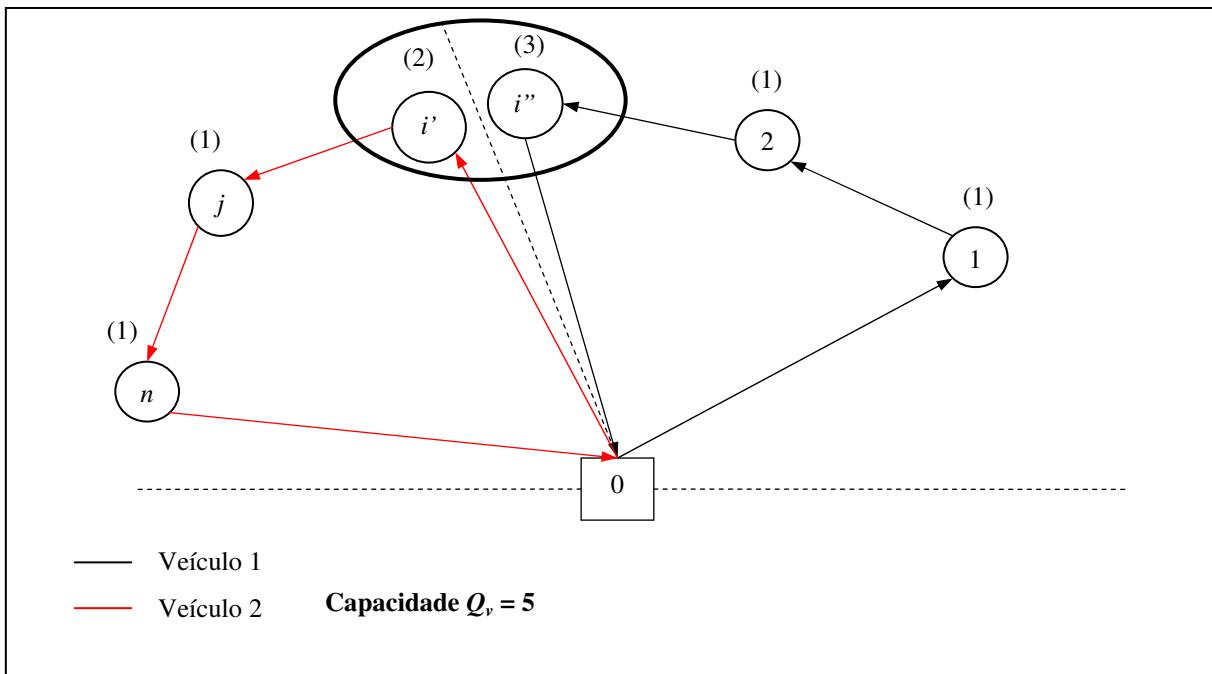


Figura 3.2.5 – O ponto  $i$  não pode ser atendido pelo veículo 1 no modelo original do método de varredura

O exemplo da figura 3.2.5, mostra os roteiros formados pelo algoritmo original. Nesta situação, o cliente  $i$  não pode ser atendido pelo veículo 1, pois sua demanda é maior que a capacidade restante do veículo 1 após este atender o ponto 2 (no caso, essa capacidade restante é igual a 3, uma vez que este veículo atendeu os pontos 1 e 2 de demanda igual a 1 unidade cada). Dessa forma, são necessários 3 veículos para atender todos os pontos do problema.



*Figura 3.2.6 – Adaptação do método de varredura*

A figura 3.2.6 mostra a adaptação do modelo, com o ponto  $i$  sendo dividido em dois pontos imaginários  $i'$  e  $i''$ , na mesma posição do ponto  $i$  original, onde a demanda de  $i'$  é igual à capacidade restante do veículo 1 (pois a varredura está sendo executada no veículo 1), ou seja, 3 unidades. O ponto  $i''$  recebe o restante da demanda de  $i$  (2 unidades) e torna-se o primeiro ponto da varredura do veículo 2. Dessa forma a demanda do ponto  $i$  fica dividida entre os veículos 1 e 2 (3 unidades para o veículo 1 e 2 unidades para o veículo 2). Portanto, são necessários apenas dois veículos para atender todos os pontos.

Como é possível observar nas figuras 3.2.5 e 3.2.6, além de possibilitar o fracionamento de entregas, esta adaptação também possibilita economias em termos de número de veículos utilizados e um melhor aproveitamento de suas capacidades.

### 3.3. EXEMPLO REDUZIDO

Esta seção apresenta um exemplo reduzido, extraído de uma situação real da empresa, com o intuito de mostrar uma aplicação prática da metodologia previamente proposta neste capítulo, ou seja, o algoritmo da varredura de Gillett e Miller para a resolução de um problema de roteirização com entregas fracionadas.

O tópico será dividido em três partes. A primeira parte introduz os dados iniciais do problema a ser resolvido. Na segunda parte, será apresentada de maneira sucinta a resolução exata do exemplo considerado através do método iterativo descrito na seção 3.1. A terceira parte mostra a resolução do problema com a aplicação do algoritmo de varredura. Por fim, as soluções obtidas em ambos os métodos serão comparadas a fim de verificar a eficiência do algoritmo proposto.

A plataforma utilizada para a resolução dos problemas foi o *software* de otimização “*What’s Best!*” da empresa *LINDO Systems*. A grande vantagem deste software em relação aos demais de sua categoria é a interface com o *software* editor de planilhas *Microsoft Excel*, uma aplicação bastante difundida, de fácil acesso para qualquer empresa e com uma interface amigável com o usuário.

#### 3.3.1. Dados do problema

No problema reduzido a seguir, retirado de uma operação real de entrega de mercadorias, há 15 pontos de demanda (numerados de 1 a 15), mais o depósito central, representado pelo ponto 0 (zero). Os dados do problema, como a distância e das demandas dos clientes foram simplificados para facilitar o entendimento. A seguir são apresentados os dados do problema necessários para sua resolução.

Neste exemplo, as cargas são consideradas em termos de caixas (volumes). Além disso, a empresa conta com dois veículos para realizar as entregas, sendo que cada um deles possui capacidade de carga igual a 60 caixas. A tabela 3.3.1 mostra os dados referentes aos clientes.

Tabela 3.3.1 – Demanda e localização dos clientes

Ponto	Local	Coord x	Coord y	Demanda
<b>0</b>	Depósito	0	0	
<b>1</b>	Loja 1	1,1	-10,7	<b>5</b>
<b>2</b>	Loja 2	0,6	-15,1	<b>2</b>
<b>3</b>	Loja 3	-1,0	-3,1	<b>63</b>
<b>4</b>	Loja 4	34,7	11,9	<b>7</b>
<b>5</b>	Loja 5	-10,1	-9,5	<b>2</b>
<b>6</b>	Loja 6	-10,5	-9,5	<b>4</b>
<b>7</b>	Loja 7	-2,9	-14,4	<b>10</b>
<b>8</b>	Loja 8	-20,9	1,5	<b>4</b>
<b>9</b>	Loja 9	-6,1	-0,7	<b>2</b>
<b>10</b>	Loja 10	-5,9	-1,7	<b>1</b>
<b>11</b>	Loja 11	-10,8	-11,4	<b>1</b>
<b>12</b>	Loja 12	18,2	-19,6	<b>15</b>
<b>13</b>	Loja 13	-0,9	-6,5	<b>1</b>
<b>14</b>	Loja 14	23,1	-18,0	<b>1</b>
<b>15</b>	Loja 15	-11,4	-8,5	<b>1</b>

A figura 3.3.1 mostra a disposição geográfica dos pontos.

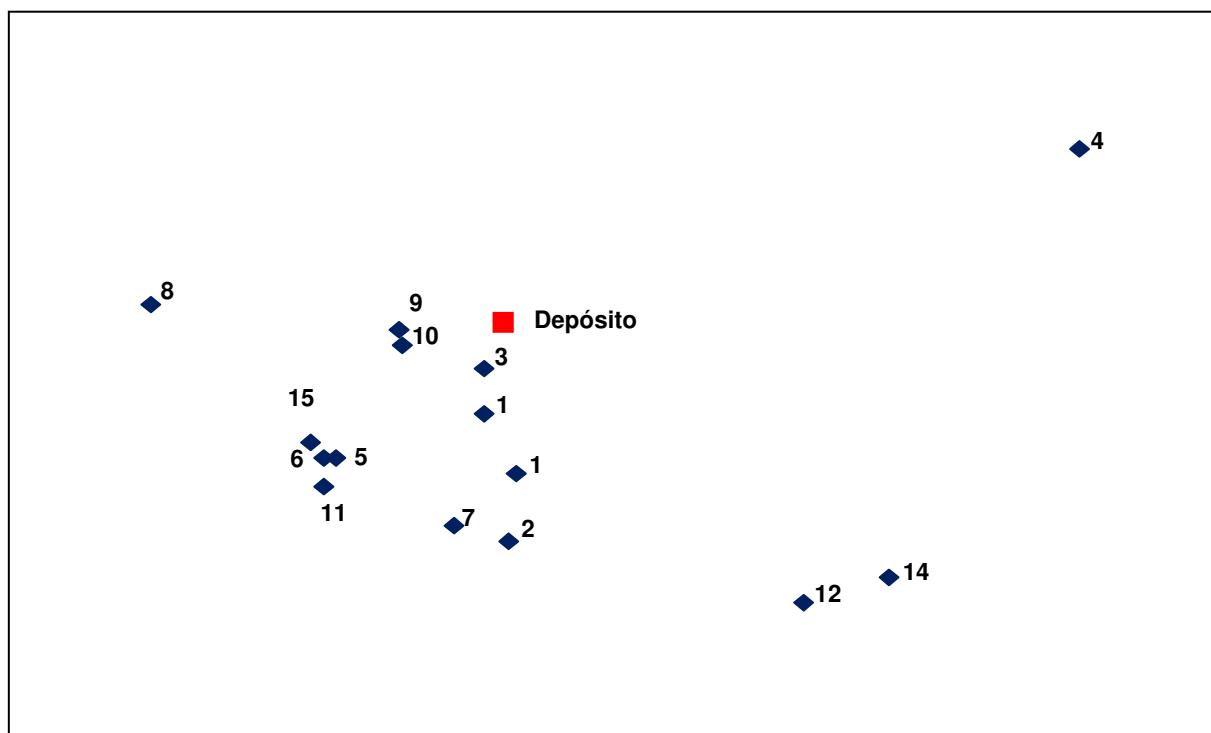


Figura 3.3.1 – Disposição geográfica dos pontos

A matriz de distâncias é mostrada na tabela 3.3.2.

*Tabela 3.3.2 – Matriz de distâncias entre os pontos do problema*

De ( <i>i</i> )	Para ( <i>j</i> )															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	0	6,0	8,4	1,8	18,9	7,4	7,5	8,1	10,7	3,1	3,1	8,4	14,3	3,6	15,5	7,5
1	6,0	0	2,5	4,3	21,3	5,8	6,0	2,9	13,1	6,6	6,1	6,1	10,0	2,5	11,9	6,5
2	8,4	2,5	0	6,7	23,0	6,3	6,5	1,8	14,3	8,7	8,2	6,2	9,3	4,9	11,6	7,2
3	1,8	4,3	6,7	0	20,0	5,9	6,0	6,3	10,5	2,9	2,6	6,8	13,4	1,9	14,8	6,1
4	18,9	21,3	23,0	20,0	0	25,8	26,0	24,1	29,0	22,0	22,0	26,6	19,5	20,8	17,7	26,1
5	7,4	5,8	6,3	5,9	25,8	0	0,2	4,6	8,2	5,3	4,9	1,1	15,5	5,0	17,6	0,9
6	7,5	6,0	6,5	6,0	26,0	0,2	0	4,7	8,1	5,4	5,0	1,1	15,7	5,2	17,8	0,7
7	8,1	2,9	1,8	6,3	24,1	4,6	4,7	0	12,7	7,7	7,2	4,3	11,1	4,5	13,4	5,4
8	10,7	13,1	14,3	10,5	29,0	8,2	8,1	12,7	0	7,7	7,9	8,9	23,1	11,1	24,9	7,4
9	3,1	6,6	8,7	2,9	22,0	5,3	5,4	7,7	7,7	0	0,5	6,4	16,2	4,1	17,7	5,1
10	3,1	6,1	8,2	2,6	22,0	4,9	5,0	7,2	7,9	0,5	0	6,0	15,8	3,7	17,4	4,7
11	8,4	6,1	6,2	6,8	26,6	1,1	1,1	4,3	8,9	6,4	6,0	0	15,4	5,7	17,7	1,7
12	14,3	10,0	9,3	13,4	19,5	15,5	15,7	11,1	23,1	16,2	15,8	15,4	0	12,1	2,7	16,3
13	3,6	2,5	4,9	1,9	20,8	5,0	5,2	4,5	11,1	4,1	3,7	5,7	12,1	0	13,8	5,5
14	15,5	11,9	11,6	14,8	17,7	17,6	17,8	13,4	24,9	17,7	17,4	17,7	2,7	13,8	0	18,4
15	7,5	6,5	7,2	6,1	26,1	0,9	0,7	5,4	7,4	5,1	4,7	1,7	16,3	5,5	18,4	0

### 3.3.2. Solução exata do exemplo reduzido

Esta seção apresenta a resolução do exemplo apresentado no item anterior através do procedimento iterativo descrito no tópico 3.1. A solução exata para o problema, encontrada por intermédio deste procedimento, servirá como parâmetro de comparação para o modelo heurístico de varredura proposto no presente trabalho. A resolução será apresentada sucintamente, uma vez que a metodologia de aplicação do procedimento iterativo já foi descrita detalhadamente no item 3.1.

A solução inicial do problema, resolvido com o auxílio do software “*What’s Best!*”, possui uma distância total de **77,8** km. Esta solução apresenta a configuração mostrada na tabela 3.3.3.

Tabela 3.3.3 – Solução inicial do exemplo reduzido

<b>Veículo</b>	<b>Roteiros formados</b>
1	0-3-13-0 4-14-12-4
2	0-3-0 1-7-2-1 6-15-11-5-6 8-9-10-8

Pode-se observar que, na solução apresentada na tabela 3.3.3, houve a formação de 4 subrotas, a saber: 4-14-12-4 no roteiro do veículo 1 e 1-7-2-1; 6-15-11-5-6 e 8-9-10-8, no roteiro do veículo 2. Escolheu-se então a subrota 8-9-10-8 para ser eliminada. Dessa forma, aplica-se as restrições de eliminação de subrotas apresentadas a seguir.

$$x_{8,9}^1 + x_{9,10}^1 + x_{10,8}^1 \leq 2$$

$$x_{8,9}^2 + x_{9,10}^2 + x_{10,8}^2 \leq 2$$

$$x_{8,10}^1 + x_{10,9}^1 + x_{9,8}^1 \leq 2$$

$$x_{8,10}^2 + x_{10,9}^2 + x_{9,8}^2 \leq 2$$

O processo se repete até que não haja nenhuma subrota nos roteiros dos veículos.

A tabela 3.3.4 apresenta um resumo das iterações realizadas até a obtenção da solução ótima do problema. Na tabela, estão compilados os custos de viagem em cada iteração, bem como os roteiros e subrotas formados, além da subrota selecionada para ser eliminada em cada iteração.

Tabela 3.3.4 – Resumo das iterações da resolução exata do exemplo reduzido

Iteração	Solução	Veículo	Rotas	Subrotas	Subrota a ser eliminada
<b>0</b>	<b>77,8</b>	1	0-3-13-0	4-14-12-4	<b>1-7-2-1</b>
		2	0-3-0	<b>1-7-2-1</b> 6-15-11-5-6 8-9-10-8	
<b>1</b>	<b>78,2</b>	1	0-3-0	1-2-7-13-1 5-6-15-11-5	<b>8-9-10-8</b>
		2	0-3-0	<b>8-9-10-8</b> 4-14-12-4	
<b>2</b>	<b>79,0</b>	1	0-3-0		
		2	0-10-9-8-15-6-5-11-7-2-1-13-3-0	<b>4-14-12-4</b>	<b>4-14-12-4</b>
<b>3</b>	<b>85,8</b>	1	0-4-14-12-2-7-1-13-3-0		<b>5-11-6-5</b>
		2	0-3-0	<b>5-11-6-5</b>	
<b>4</b>	<b>85,9</b>	1	0-9-10-3-0	<b>5-6-15-8-11-5</b>	<b>5-6-15-8-11-5</b>
		2	0-4-14-12-2-7-1-13-3-0		
<b>6</b>	<b>86,3</b>	1	0-3-0	<b>5-11-15-6-5</b>	<b>5-11-15-6-5</b>
		2	0-4-14-12-2-7-1-13-3-10-8-9-0		
<b>7</b>	<b>86,4</b>	1	0-3-0	<b>5-11-6-15-8-9-10-5</b>	<b>5-11-6-15-8-9-10-5</b>
		2	0-4-14-12-2-7-1-13-3-0		
<b>8</b>	<b>86,5</b>	1	0-3-0	Nenhuma	Nenhuma
		2	0-4-14-12-1-2-7-11-5-6-15-8-9-10-13-3-0		

Conforme mostra a tabela 3.3.4, a solução ótima do problema apresenta um custo total de **86,5** e foi obtida após 8 iterações, com a eliminação de 7 subrotas. Observa-se que na solução final, o cliente 3, que possui demanda igual a 63 unidades, maior que a capacidade máxima dos veículos, igual a 60 volumes tem sua demanda dividida entre os veículos 1 e 2. No caso, o veículo 1 atende apenas o cliente 3, ao passo que o veículo 2 atende os demais,

inclusive a demanda restante do cliente 3, que o veículo 1 não conseguiu atender. A tabela 3.3.5 mostra as frações de demanda de cada cliente atendidas pelos veículos.

*Tabela 3.3.5 – Frações de demanda atendidas pelos veículos*

Cliente	Demanda	Veículo 1		Veículo 2	
		Fração de demanda	Demandas atendidas	Fração de demanda	Demandas atendidas
1	5	0	0	1	5
2	2	0	0	1	2
3	63	0,95	60	0,05	3
4	7	0	0	1	7
5	2	0	0	1	2
6	4	0	0	1	4
7	10	0	0	1	10
8	4	0	0	1	4
9	2	0	0	1	2
10	1	0	0	1	1
11	1	0	0	1	1
12	15	0	0	1	15
13	1	0	0	1	1
14	1	0	0	1	1
15	1	0	0	1	1
			60		59

Para a obtenção da solução ótima sem subrotas, foi necessária a inclusão de 61 restrições de eliminação, a partir da solução inicial do problema relaxado (isto é, sem nenhuma restrição de eliminação de subrotas). Caso o problema tivesse sido formulado conforme o modelo original de Dror, Laporte e Trudeau (1994), o número total de restrições de eliminação de subrotas seria da ordem de 32.000. Este exemplo ilustrativo mostra que através dessa abordagem é possível obter soluções exatas de problemas que, resolvidos pela formulação original consumiriam muito mais tempo e recursos computacionais.

### 3.3.3. Resolução através do método de varredura

Esta seção apresenta a resolução do exemplo reduzido através do método heurístico de varredura, descrito no item 3.2, com o intuito de ilustrar a aplicação do modelo em um problema real.

No primeiro passo do algoritmo de varredura, os pontos devem ser classificados em ordem crescente de seus ângulos de coordenadas polares ( $\theta_i, r_i$ ) em relação ao eixo  $x$  de

coordenadas cartesianas. O ângulo de coordenada polar  $\theta_i$  de um determinado ponto  $i$  é calculado através da equação (3.3) apresentada anteriormente. Dessa forma, os pontos do problema foram reclassificados conforme a tabela 3.3.6 a seguir.

*Tabela 3.3.6 – Reclassificação dos pontos em ordem crescente de ângulo de coordenada polar*

<b>Classificação</b>	<b>Local</b>	<b>Coordenadas cartesianas</b>		<b>Coordenadas polares</b>		<b>Demand</b>
		<b>Coord x</b>	<b>Coord y</b>	<b>Ângulo (<math>\theta_i</math>)</b>	<b>Raio (<math>r_i</math>)</b>	
<b>0</b>	Depósito	0	0			
<b>1</b>	Loja 4	34,7	11,9	0,331	36,7	7
<b>2</b>	Loja 8	-20,9	1,5	3,072	21,0	4
<b>3</b>	Loja 9	-6,1	-0,7	3,263	6,1	2
<b>4</b>	Loja 10	-5,9	-1,7	3,421	6,1	1
<b>5</b>	Loja 15	-11,4	-8,5	3,780	14,3	1
<b>6</b>	Loja 6	-10,5	-9,5	3,874	14,2	4
<b>7</b>	Loja 5	-10,1	-9,5	3,894	13,9	2
<b>8</b>	Loja 11	-10,8	-11,4	3,957	15,7	1
<b>9</b>	Loja 3	-1,0	-3,1	4,417	3,3	63
<b>10</b>	Loja 7	-2,9	-14,4	4,512	14,7	10
<b>11</b>	Loja 13	-0,9	-6,5	4,577	6,6	1
<b>12</b>	Loja 2	0,6	-15,1	4,754	15,1	2
<b>13</b>	Loja 1	1,1	-10,7	4,815	10,7	5
<b>14</b>	Loja 12	18,2	-19,6	5,459	26,7	15
<b>15</b>	Loja 14	23,1	-18,0	5,622	29,3	1

Dessa forma, a varredura iniciará pelo ponto correspondente à Loja 4, ou seja, o ponto de número 4 conforme classificação original. Para evitar ambigüidades, os pontos reclassificados não terão os seus números originais trocados no mapa. Portanto, essa classificação atuará apenas na determinação do ponto de início da varredura. A figura 3.3.2 ilustra o início do procedimento de varredura, partindo do ponto inicial (Loja 4).

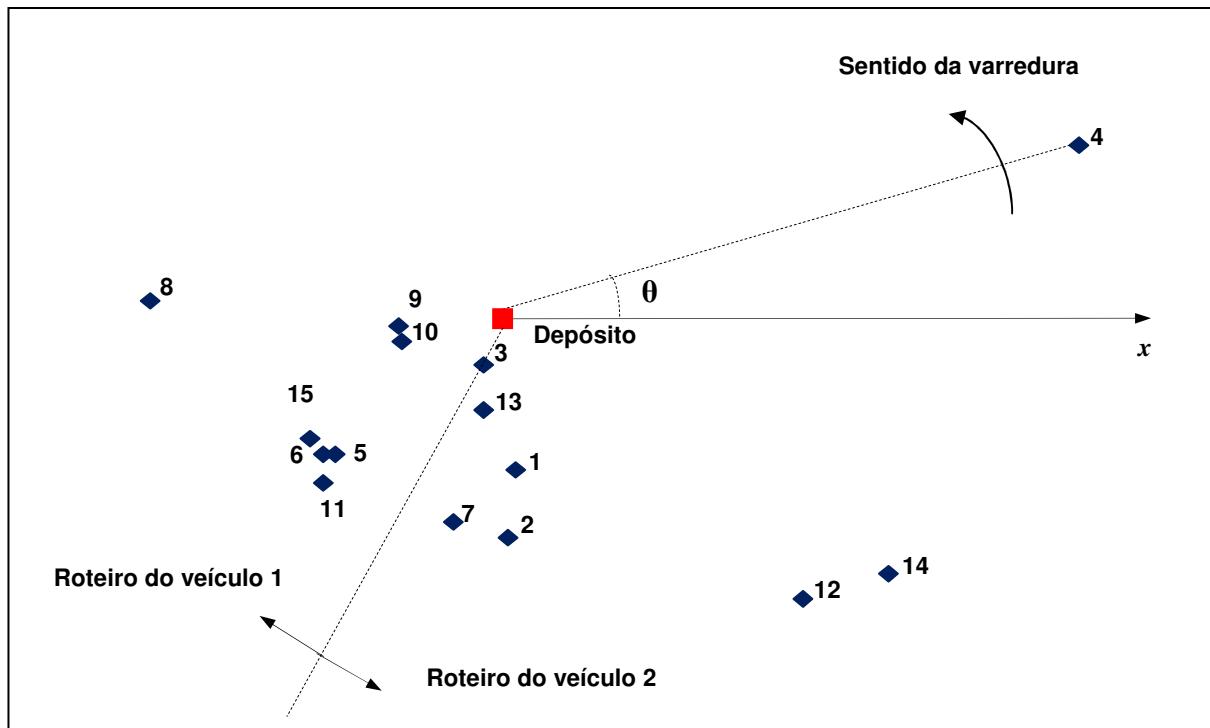


Figura 3.3.2 – Início do procedimento de varredura

A tabela 3.3.7 mostra a demanda acumulada dos clientes reclassificados no passo anterior. Enquanto esta demanda acumulada for menor que a capacidade do veículo  $v$ , este veículo pode atender o próximo ponto.

*Tabela 3.3.7 – Demanda acumulada dos clientes e alocação dos veículos*

<b>Classificação</b>	<b>Local</b>	<b>Demand</b>	<b>Demand acumulada</b>	<b>Veiculo</b>
0	Depósito			
1	Loja 4	7	7	1
2	Loja 8	4	11	1
3	Loja 9	2	13	1
4	Loja 10	1	14	1
5	Loja 15	1	15	1
6	Loja 6	4	19	1
7	Loja 5	2	21	1
8	Loja 11	1	22	1
<b>9</b>	<b>Loja 3</b>	<b>63</b>	<b>85</b>	<b>1 e 2</b>
10	Loja 7	10	95	2
11	Loja 13	1	96	2
12	Loja 2	2	98	2
13	Loja 1	5	103	2
14	Loja 12	15	118	2
15	Loja 14	1	119	2

Durante a varredura do primeiro veículo, observa-se que no ponto 9, de demanda 63, a demanda acumulada ultrapassa a capacidade do veículo (que é de 60 unidades). No algoritmo original, o veículo 1 não teria condições de atender o ponto 9 e o último ponto que esse veículo poderia atender é o ponto 8 (Loja 11). Na realidade, neste caso o problema não poderia ser resolvido com o algoritmo original, pois a demanda de um único ponto ultrapassa a capacidade dos veículos.

Com a adaptação do problema para permitir entregas fracionadas, a demanda da Loja 3 (isto é o ponto 9) é dividida entre os veículos 1 e 2, onde a demanda atendida pelo veículo 2 é de  $85 - 60 = 25$  volumes, e a demanda atendida pelo veículo 1 é igual a  $60 - 22 = 38$  volumes.

Nota-se que o conceito envolvido neste fracionamento de demanda está relacionado à transformação do ponto 9 em dois pontos imaginários, 9' e 9'', de tal maneira que o veículo 1 atenda o ponto 9' e o veículo 2 o ponto 9''. A demanda do ponto 9' é igual à capacidade restante do veículo após este atender o ponto imediatamente anterior ao ponto 9'. Já a demanda do ponto 9'' é igual à demanda original do ponto 9 menos a demanda de ponto 9'.

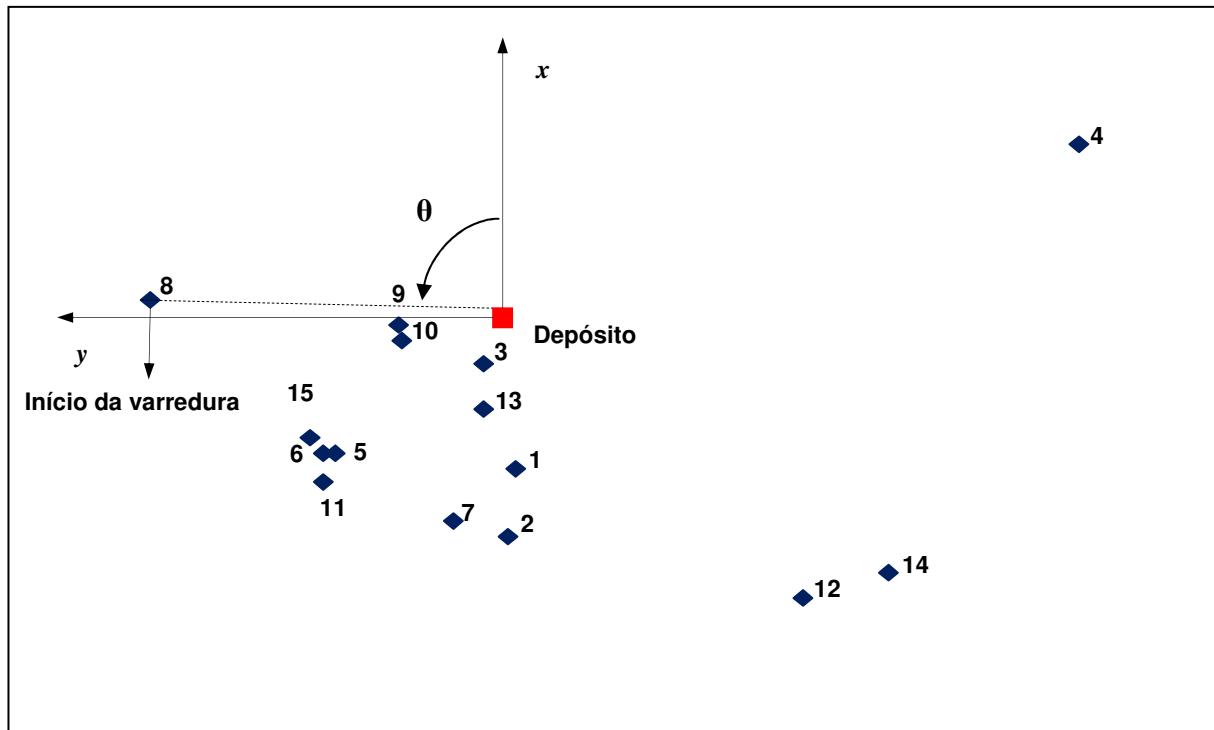
Após a atribuição dos pontos aos veículos através da varredura, os roteiros de cada veículo são determinados através da resolução de um problema de caixeiro viajante.

Conforme mencionado anteriormente, os problemas de caixeiro viajante serão resolvidos através da aplicação da formulação de Miller, Tucker e Zemlin (1960), apresentada no Apêndice A, utilizando o *software* de otimização “*What's Best!*” e modelados no aplicativo *Microsoft Excel*. A tabela 3.3.8 mostra a alocação dos pontos aos veículos após a primeira varredura e o custo da solução do problema de caixeiro viajante para cada veículo.

*Tabela 3.3.8 - Rotas dos veículos após a primeira varredura*

Veículo	Rota	Custo
1	0-4-3-5-11-6-15-8-9-10-0	66,4
2	0-3-13-1-7-2-12-14-0	38,4

A segunda varredura é aplicada após a rotação de 90° no sentido anti-horário do eixo ( $x,y$ ), em relação à posição inicial. A figura 3.3.3 ilustra este procedimento.



*Figura 3.3.3 – Rotação do eixo ( $x,y$ ) para execução da segunda varredura*

O processo se repete até que o eixo retorne à sua posição inicial após as rotações. Depois das 3 rotações possíveis, obteve-se os resultados da tabela 3.3.9.

Tabela 3.3.9 – Resultados do método de varredura

<b>Varredura</b>	<b>Veículo</b>	<b>Rota</b>	<b>Custo</b>	<b>Total</b>
1	1	0-4-3-5-11-6-15-8-9-10-0	66,4491	104,8
	2	0-3-13-1-7-2-12-14-0	38,3997	
2	1	0-3-5-11-6-15-8-9-10	29,3271	<b>88,8</b>
	2	0-4-14-12-2-7-1-13-3-0	59,4955	
3	1	0-3-11-5-6-15-10-9-0	19,0007	97,9
	2	0-4-14-12-2-7-1-13-3-8-0	78,9052	
4	1	0-13-8-7-6-9-10-11-12-2-1-3-4-5-0	79,1439	95,4
	2	0-13-7-3-0	16,2746	

Da tabela 3.3.9 pode-se observar que a solução obtida através do método de varredura apresenta um custo total de **88,82** com a demanda do cliente 3 sendo dividida entre os dois veículos. A figura 3.3.4 apresenta os roteiros formados nesta solução.

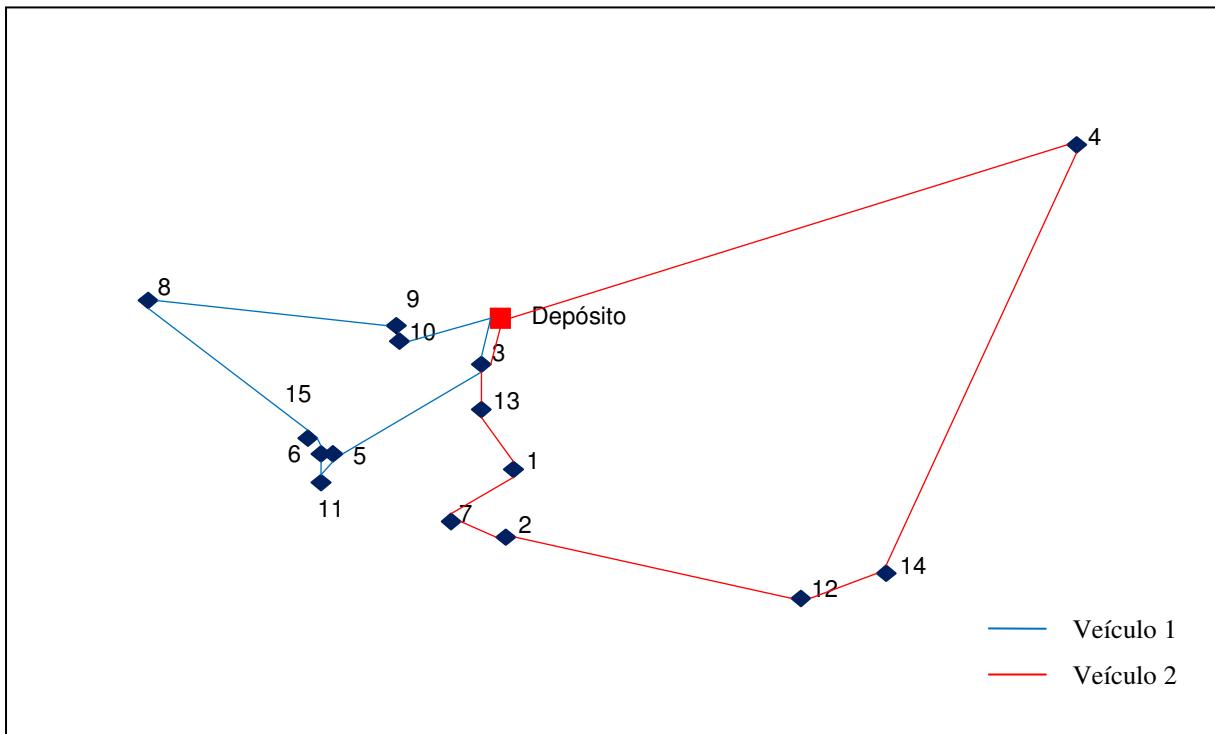


Figura 3.3.4 – Solução obtida através do método de varredura

Os resultados obtidos, tanto com a utilização do método exato (procedimento iterativo) quanto pelo método heurístico (algoritmo de varredura) serão então comparados com a

situação que efetivamente ocorreu neste caso (isto é com os roteiros reais gerados pela empresa). A tabela 3.3.10 apresenta a comparação das soluções obtidas com a situação real.

*Tabela 3.3.10 – Comparaçao das soluções obtidas com a situação real*

	<b>Veículo</b>	<b>Roteiro</b>	<b>Distância</b>
<b>Situação real</b>	1	0-3-0 (1 <sup>a</sup> . viagem) 0-4-3-10-8-9-0 (2 <sup>a</sup> . viagem)	<b>109,6</b>
	2	0-13-1-14-12-2-7-11-5-6-15-0	
<b>Solução exata</b>	1	0-3-0	<b>86,5</b>
	2	0-4-14-12-1-2-7-11-5-6-15-8-9-10-13-3-0	
<b>Solução heurística</b>	1	0-3-5-11-6-15-8-9-10-0	<b>88,8</b>
	2	0-4-14-12-2-7-1-13-3-0	

Comparando as soluções obtidas através do método exato e através do método de varredura com a situação real, observa-se que a economia gerada foi razoável, apresentando um percentual de economia de aproximadamente 21% no primeiro caso e 18% no segundo. Além disso, pode-se observar que a solução obtida com a aplicação do algoritmo é apenas 2,5% maior que a solução ótima do problema.

Vale ressaltar que na situação real, um dos veículos (no caso o veículo 1) realizou duas viagens no dia considerado. Na primeira delas, o veículo foi enviado com carga completa para atender exclusivamente ao cliente 3 cuja demanda é superior a capacidade dos veículos. Na segunda viagem realizada, o veículo 1 atendeu o restante da demanda do cliente 3 e também os outros pontos para os quais havia sido previamente designado.

Os problemas de roterização de múltiplas viagens, que envolvem retornos ao depósito, por exemplo, no caso em que a soma de todas as demandas dos clientes é maior que a soma das capacidades de carga dos veículos disponíveis, estão fora do escopo deste trabalho. No entanto, a possibilidade de realização de múltiplas viagens pode ser considerada para um refinamento do modelo proposto em trabalhos futuros.

## 4. LEVANTAMENTO E COLETA DE DADOS

Este capítulo discorre acerca dos dados necessários para a elaboração do modelo de roteirização proposto neste trabalho. A primeira parte apresenta todos os dados relevantes para o problema considerado, necessários para a resolução do modelo em si. A segunda parte trata do processo de obtenção dos dados, apresentando a metodologia de levantamento dos dados e as fontes, nas quais eles foram obtidos. Finalmente, a terceira parte apresenta a compilação dos dados obtidos de acordo com as necessidades do modelo.

### 4.1. DADOS DO PROBLEMA

Os dados de entrada de um problema de roteirização de veículos, basicamente estão associados aos parâmetros definidos na formulação do problema. No caso da formulação de Dror, Laporte e Trudeau (1994) para o SDVRP, os parâmetros relevantes são a capacidade dos veículos, as demandas de cada cliente e os custos de viagem entre dois pontos. Pode-se então dividir os dados necessários em três categorias distintas, a saber, dados sobre os veículos, dados sobre os clientes e dados sobre a distribuição.

#### Dados sobre os veículos

- Capacidade de carga dos veículos;
- Número de veículos que a empresa dispõe.

#### Dados dos clientes

- Localização (endereço) dos clientes;
- Custos (distâncias) de viagem de um ponto  $i$  a um ponto  $j$ .

#### Dados da distribuição

- Clientes que serão atendidos em determinado dia;
- Demanda dos clientes que serão atendidos em determinado dia.

### 4.2. OBTENÇÃO DOS DADOS

Os dados necessários para a resolução do problema, apresentados na seção anterior, basicamente foram coletados de duas formas: diretamente com a empresa estudada e através de fontes externas. Após a coleta dos dados brutos, foi necessário um tratamento dessas informações de modo a possibilitar a entrada dos mesmos no modelo proposto.

#### 4.2.1. Dados sobre os veículos

Os dados sobre os veículos são de fácil obtenção e foram coletados diretamente da empresa, através de entrevistas com os encarregados do transporte e distribuição de mercadorias. Conforme mencionado no primeiro capítulo, o transporte de mercadorias do armazém da empresa (TRADBRAS) aos clientes é realizado através de *vans*, disponibilizadas por uma empresa contratada.

A tabela 4.2.1 a seguir apresenta as principais informações dos veículos obtidas na empresa.

*Tabela 4.2.1 – Informações gerais sobre os veículos*

<b>Informações sobre os veículos</b>		
Número de veículos		2
Motoristas por veículo		1
Ajudantes por veículo		1
Jornada de trabalho (horas/dia)		8
Capacidade de carga do veículo	(t) (caixas)	1,5 60

Vale ressaltar que outras informações operacionais dos veículos, como velocidade média, consumo de combustível, entre outros, não fazem parte do escopo do problema abordado, porém são de grande utilidade para a elaboração de uma estrutura de custos mais detalhado, que poderia vir a ser utilizada em um refinamento do modelo proposto.

#### 4.2.2. Dados sobre os clientes

Basicamente, a coleta de dados dos clientes tem como objetivo a formação de um registro de todos os clientes considerados na resolução de problema (isto é, os clientes da região da Grande São Paulo), bem como a determinação das distâncias de viagem de um ponto *i* a outro ponto *j*, com base nos endereços (posição geográfica) de cada cliente.

A empresa conta com um banco de dados com o cadastro de todos os clientes (ativos e inativos), de onde foi possível obter os endereços de cada um deles. Por se tratar de um banco de dados completo, com uma série de informações específicas de cada cliente, tornou-se

necessária uma filtragem nos dados de saída deste cadastro, apenas com a seleção das informações necessárias para a resolução do problema. Dessa forma, foi criado um cadastro interno dos clientes, na forma de planilha eletrônica, contendo os dados relevantes para a resolução do problema, ou seja, os códigos (utilizados pela empresa), os nomes e endereços dos clientes.

### Cálculo das distâncias de viagem

Conforme mencionado anteriormente no primeiro capítulo (item 1.1 – “Descrição da empresa”), a TRADBRAS tem em seu cadastro mais de 1.000 clientes, com 500 deles localizados na região da Grande São Paulo. Além disso, a rotina de entregas é realizada em base diária, de acordo com a demanda dos clientes, com um diferente conjunto de pontos (clientes) a serem atendidos a cada dia.

Assim sendo, o cálculo de distâncias individuais entre cada par de endereços é inviável neste caso, onde há 500 endereços a serem verificados, e cujo processo de entregas é dinâmico, ou seja, muda a cada dia. Dessa forma, procedeu-se a uma ferramenta de cálculo para a determinação das distâncias entre dois pontos, com base em suas coordenadas.

O método de cálculo de distâncias em sua forma elementar envolve a aplicação de trigonometria simples, em um sistema de coordenadas cartesianas ( $x,y$ ). No entanto, segundo Ballou (2005), distorções causadas por técnicas de mapeamento na projeção do globo terrestre em um plano, podem ocasionar erros significativos, no caso em que as distâncias são calculadas através do método simples de grade linear.

Uma abordagem que produz resultados mais próximos dos valores reais é a equação de distância de grande círculo, que envolve a aplicação de trigonometria esférica e transformações de coordenadas esféricas ( $\rho,\theta,\varphi$ ) em cartesianas ( $x,y$ ). Além de minimizar tais distorções de mapeamento, este modelo considera adequadamente a curvatura da Terra. A equação (4.1) apresenta a fórmula de cálculo da distância de grande círculo.

$$Dist_{A-B} = R * \{ \arccos[\sin(LAT_A) * \sin(LAT_B) + \cos(LAT_A) * \cos(LAT_B) * \cos(|LONG_B - LONG_A|)] \} \quad (4.1)$$

Onde:

$Dist_{A-B}$  = distância de grande círculo do ponto A ao ponto B.

$R$  = raio de curvatura da Terra, igual a 6.377 km

$LAT_A$  = latitude do ponto A (em radianos)

$LAT_B$  = latitude do ponto B (em radianos)

$LONG_A$  = longitude do ponto A (em radianos)

$LONG_B$  = longitude do ponto B (em radianos)

Vale ressaltar que as coordenadas de pontos, em latitudes e longitudes encontradas em mapas e ferramentas de localização são dadas em graus ( $^{\circ}$ ). Dessa forma, para a entrada das coordenadas na equação (4.1), é necessário realizar a conversão dos valores das coordenadas em graus para radianos de acordo com a seguinte relação:

$$\frac{\text{Valor da coordenada em radianos}}{\text{coordenada em graus}} = \frac{\pi}{180} \quad (4.2)$$

As coordenadas dos endereços dos pontos de entrega, em latitudes e longitudes, foram extraídas do serviço *web* gratuito de mapas e rotas *Google Maps* ([www.maps.google.com](http://www.maps.google.com)), da empresa *Google*, líder em serviços e soluções de internet. Através da entrada do endereço do local desejado, é possível obter as coordenadas do ponto localizado através do próprio *link* para o qual o usuário é redirecionado após entrar com o endereço desejado.

Ballou (2005) considera este modelo bastante adequado para determinar a distância entre dois pontos, principalmente pela boa precisão nos cálculos, bem como pela facilidade de obter as coordenadas dos pontos em latitudes e longitudes, encontrados em diversas fontes, como mapas e serviços de localização. No entanto, por calcular as distâncias em linha reta, este modelo não considera adequadamente fatores como curvas, desvios e retornos que um veículo pode realizar enquanto trafega de um ponto a outro. Dessa forma, as distâncias calculadas podem ser ajustadas através de um fator de sinuosidade (ou fator médio de circuito), que considera todos os fatores supramencionados. A tabela 4.2.2, (adaptada de BALLOU, 2005), mostra fatores de sinuosidades obtidos empiricamente em alguns países selecionados.

*Tabela 4.2.2 – Fatores médios de circuito em alguns países – Adaptado de Ballou (2005)*

<b>País</b>	<b>Número de pontos</b>	<b>Fator médio de circuito</b>	<b>Desvio-padrão</b>
Argentina	66	1,22	0,15
Austrália	77	1,28	0,17
<b>Brasil</b>	<b>120</b>	<b>1,23</b>	<b>0,11</b>
Canadá	49	1,30	0,10
Estados Unidos	299	1,20	0,17
Europa	1,99	1,46	0,58
México	49	1,46	0,04

A tabela 4.2.2 mostra que o fator de circuito estimado para o Brasil é de **1,23**, com desvio-padrão de **0,11**. Este número será adotado neste trabalho como o fator pelo qual a distância de grande círculo calculada deve ser ajustada, possibilitando a determinação de um valor mais próximo da realidade. No entanto, um fator de circuito mais detalhado pode ser obtido através da comparação entre distâncias reais obtidas em uma amostra e as respectivas distâncias calculadas.

#### **4.2.3. Dados sobre a distribuição**

O processo de distribuição de mercadorias ao varejo na empresa estudada é dinâmico, ou seja, a cada dia, os endereços a serem visitados pelos veículos são diferentes. Em outras palavras, a cada dia, o problema de roterização a ser resolvido é diferente. Dessa forma, os dados necessários para entrada no modelo, referentes à operação diária de distribuição são, basicamente:

- Quantos clientes serão atendidos em determinado dia;
- Quais clientes serão atendidos em determinado dia;
- Os endereços dos clientes que serão atendidos;
- A demanda dos clientes que serão atendidos.

Estes dados foram obtidos através das informações contidas em formulários que são entregues pela empresa diariamente aos motoristas dos veículos, contendo os clientes designados a cada veículo e as quantidades demandadas por cada um desses clientes. Nota-se que a atribuição dos clientes aos veículos é feita pela empresa (TRADBRAS), no entanto, os roteiros a serem percorridos pelos veículos, são determinados pelos próprios motoristas, com base em sua experiência e bom senso.

A tabela 4.2.3 a seguir, apresenta alguns dados referentes à operação de distribuição na região da Grande São Paulo

*Tabela 4.2.3 – Dados gerais sobre a distribuição*

<b>Informações sobre a distribuição (Grande São Paulo)</b>	
Nº total de clientes	500
Nº médio de entregas por dia	25
Nº médio de viagens/dia (por veículo)	1
Localização dos clientes (%)	
Zona Sul	50%
Zona Oeste	25%
Zona Leste	15%
Zona Norte	10%

### **4.3. COMPILAÇÃO DOS DADOS**

Esta seção apresenta uma compilação dos dados obtidos, necessários para a resolução do problema considerado, com o intuito de formar uma estrutura lógica e organizada para a sua entrada no modelo proposto, tornando o processo de solução do modelo através da implementação do algoritmo o mais simples possível. Os dados coletados foram organizados conforme a estrutura a seguir.

#### **Entregas diárias**

A tabela 4.3.1 mostra um exemplo dos dados compilados de uma operação de distribuição em um determinado dia, com 15 pontos de demanda formatados para entrada no modelo de resolução. Observa-se que as informações necessárias contidas na tabela para cada cliente a ser atendido são: nome, endereço, demanda e coordenadas do ponto em latitude e longitude.

Tabela 4.3.1 – Compilação dos dados de entrega

Ponto	Local	Endereço	Lat	Long	Demanda
0	TRADBRAS S/A IMPORTACAO E EXPORTACAO	R. CARLOS DE SOUZA NAZARETH, 418	-23,5403	-46,6305	
1	GATE GOURMET LTDA	R. MARIA GARCIA UTRILLA, 453	-23,4807	-46,457	7
2	AGRO NIPPO PRODUTOS ALIMENTICIOS LTDA	AV. DR GASTAO VIDIGAL, 1946	-23,533	-46,7351	4
3	U.N. COMERCIO DE ALIMENTOS LTDA	PRACA VILABOIM, 93	-23,544	-46,6608	2
4	VILLA PAULISTANA RESTAURANTE LTDA	R. PARÁ, 210	-23,5487	-46,6599	1
5	SABATINI COMÉRCIO DE ALIMENTOS LTDA	R. PROFESSOR ARTUR RAMOS, 397	-23,5828	-46,6877	1
6	JAM WAREHOUSE COM. DE ALIM. E PROMO. LTDA	R. LOPES NETO, 308	-23,5878	-46,6832	4
7	COMERCIO DE ALIMENTOS YABANY LTDA	R.. PROF. ATILIO INOCENTI, 53	-23,5878	-46,6812	2
8	LMPR COMERCIO DE ALIMENTACAO E BEBIDAS LTDA	R. GOMES DE CARVALHO, 1165	-23,5975	-46,6843	1
9	COMERCIAL MARUKAI LTDA	R. GALVAO BUENO, 34	-23,5559	-46,6352	63
10	ART & DESIGN COM. E REP. LTDA	AL. DOS GUAINUMBIS, 32	-23,6121	-46,645	10
11	REGINA TIYOKO KAWADA YONAMINE	R. JUPITER, 218	-23,5728	-46,6349	1
12	CLAUDIO HIROYUKI MASUBUCHI	R. GENERAL CHAGAS SANTOS, 815	-23,6158	-46,6274	2
13	MINAMI VILAS BAR E PETISKARIA LTDA-ME.	R. PEDRO NICOLE, 07	-23,5936	-46,625	5
14	COOP - COOPERATIVA DE CONSUMO	AV. INDUSTRIAL, 2001	-23,6383	-46,5397	15
15	EMPORIO E QUITANDA SANTA PAULA LTDA-ME	AV. MARTIM FRANCISCO, 1071	-23,6303	-46,5149	1

### **Distâncias entre os pontos**

Para cada dia, ou seja, para cada roteirização diária programada, as distâncias entre os pontos a serem visitados, são exibidas na forma de uma matriz “de-para”, contendo as distâncias entre o ponto de origem ao ponto de destino, calculadas utilizando a fórmula (4.1) apresentada anteriormente. Um exemplo desta matriz foi apresentado no item 3.3 – “Exemplo reduzido”.

## 5. APLICAÇÃO DO MODELO

Este capítulo apresenta a etapa de aplicação e simulação do modelo proposto, norteado pela estrutura a seguir. A primeira parte apresenta a adaptação da modelagem do algoritmo de varredura à plataforma selecionada para ser utilizada neste trabalho, no caso a plataforma *Microsoft Excel*, associada ao *software* de otimização “*What’s Best!*”. Na segunda parte, será descrito o processo de validação do modelo proposto através de sua implantação e simulação com dados reais da empresa.

### 5.1. MODELAGEM DO ALGORITMO

A plataforma selecionada como ferramenta para a resolução do problema de roterização abordado é o *software* editor de planilhas *Microsoft Excel*, com o auxílio do programa de otimização “*What’s Best!*”, conforme mencionado no capítulo 3 (item 3.1). A grande vantagem do programa “*What’s Best!*” em relação dos demais de sua categoria (como por exemplo, LINGO e CPLEX), é a sua interface com o ambiente *Microsoft Excel*, bastante difundido e utilizado tanto por usuários domésticos e corporativos.

A aplicação “*What’s Best!*” basicamente é um suplemento da plataforma *Excel*, que fornece ferramentas de modelagem que possibilitam a formulação de modelos de otimização através de planilhas, dentro do próprio ambiente *Excel*. O *software* tem capacidade de resolver uma diversa gama de problemas de otimização linear, não-linear, inteira e mista, sendo de fácil aprendizagem de uso. Dessa forma, o processo de modelagem torna-se muito mais simples e ágil, mesmo no caso de problemas maiores e mais complexos (SCHUSTER; ALLEN, 1998).

No entanto, como o método de resolução proposto no presente trabalho é um algoritmo heurístico, que não garante a condição ótima da solução encontrada, torna-se necessária uma adaptação do modelo teórico para a linguagem da aplicação *Excel*, de maneira que o modelo possa ser efetivamente implantado na prática.

#### 5.1.1. Requisitos do modelo

Inicialmente, os requisitos que o modelo a ser desenvolvido deve satisfazer para atingir os objetivos propostos de resolver o problema de forma bem-sucedida devem ser levantados. Os principais requisitos que o modelo deve cumprir são apresentados a seguir.

## **1. Permitir entrada de dados de entrega diária**

O modelo a ser desenvolvido deve receber a entrada dos dados do problema e compilar as informações coletadas, organizando-as de tal maneira a possibilitar a inicialização do algoritmo. As informações necessárias para tanto são o número de clientes a serem atendidos em determinado dia, os nomes (ou códigos) desses clientes e suas correspondentes demandas. Os demais parâmetros, isto é as coordenadas dos pontos, são obtidos em um registro dos clientes, criado previamente. Após a obtenção das informações, a matriz “de-para” de distâncias ponto-a-ponto deve ser determinada e os ângulos dos pontos, em coordenadas polares, devem ser calculados.

## **2. Calcular parâmetros adicionais do problema**

Após a obtenção das informações, os parâmetros adicionais do problema, necessários à inicialização do algoritmo devem ser determinados, os quais são: a matriz “de-para” de distâncias ponto-a-ponto e os ângulos dos pontos, em coordenadas polares. Além disso, o modelo deve reclassificar os pontos de demanda do problema em ordem crescente de acordo com seus ângulos em coordenadas polares, conforme a primeira etapa do algoritmo de varredura.

## **3. Executar a varredura dos pontos**

O algoritmo a ser criado na plataforma *Excel* deve executar o procedimento de varredura dos pontos de demanda do problema a ser resolvido. Este procedimento deve ser realizado da seguinte forma: iniciando com o primeiro veículo, atribui-se pontos a este veículo, enquanto a demanda acumulada desses pontos for menor que a sua capacidade. Se a demanda do próximo ponto for maior que a capacidade do veículo, então parte da demanda deste cliente será atribuída ao veículo atual e o restante a próximo veículo. A varredura continua até que todos os pontos tenham sido percorridos.

## **4. Resolver os problemas de caixeiro viajante em cada rota definida na varredura**

Após cada rota definida na etapa de varredura, as rotas dos veículos devem ser determinadas através da resolução dos problemas de caixeiro viajante para cada um dos veículos. Estes problemas serão resolvidos através da ferramenta “*What’s Best!*” e para tanto será necessária:

- A criação de uma rotina que escreva a formulação dos problemas de caixeiro viajante para cada veículo.
- A criação de uma rotina que resolva cada um dos problemas supramencionados.
- Após a resolução de todos os problemas, deve-se obter o custo (distância) total da solução, através da soma de todas as soluções dos problemas de caixeiro viajante e os roteiros finais devem ser armazenados.

## **5. Realizar a rotação do eixo de referência ( $x,y$ ), inicializando a próxima varredura**

Após a resolução do processo de varredura, deve-se realizar a rotação do eixo de referência, de modo que os pontos sejam reclassificados novamente, de acordo com a rotação estabelecida, inicializando a varredura seguinte. No modelo proposto, a rotação do eixo de referência será de 90° em relação à configuração anterior, limitando a quantidade de configurações de eixos a quatro possibilidades.

## **6. Repetir o processo de varredura até que todas as configurações de eixos sejam testadas**

O processo de varredura deve se repetir até que todas as configurações de eixos tenham sido executadas de acordo com a rotação do eixo de referência.

## **7. Finalizar a execução fornecendo a solução obtida**

A execução do modelo é finalizada após a realização de todas as iterações realizadas pelo algoritmo e deve fornecer a solução final do problema, que é a solução de menor custo entre todas as obtidas em cada iteração. Os roteiros finais dos veículos também devem ser exibidos.

Assim colocado, seguir-se-á a seguinte estrutura para a tradução do modelo heurístico de varredura ao ambiente *Excel*.

## Início do algoritmo

### Inicialização

Entrada de dados: número de clientes, nomes (ou códigos) e demandas

Calcular parâmetros: ângulos de coordenadas polares e matriz de distâncias

Ordenar pontos em ordem crescente de ângulos para primeira varredura

## Varredura

**Enquanto** houver alguma configuração de início de varredura ainda não testada **faça**

Iniciar varredura com o primeiro veículo

**Enquanto** houver pontos ainda não atendidos **faça**

**Se** a demanda acumulada dos pontos atendidos pelo veículo atual é menor que a capacidade do veículo **então**

Adicionar próximo ponto ao roteiro do veículo atual

**Se** a demanda acumulada mais a demanda do próximo ponto é maior que a capacidade do veículo atual **então**

Adicionar próximo ponto ao roteiro do veículo atual

Pegar próximo veículo e continuar varredura no ponto atual

Adicionar ponto atual ao roteiro do veículo

Fim **se**

Fim **se**

Fim **enquanto**

Aplicar rotina de resolução do problema de caixeiro viajante

Armazenar resultado da solução dos problemas de caixeiro viajante

Rotacionar eixo de referência (definir novo ponto de início da varredura)

Fim **enquanto**

## Resultados

- Selecionar a solução de menor custo total obtida no passo anterior
- Exibir custo total da solução final
- Exibir roteiro da solução final

## Fim algoritmo

*Figura 5.1.1 – Descrição do algoritmo*

A estrutura apresentada na figura 5.1.1 foi traduzida para o ambiente *Excel* através da elaboração de um programa (macro) na linguagem VBA (*Visual Basic for Applications*), que funciona em conjunto com o aplicativo *Microsoft Excel*.

Este programa processa os dados iniciais do problema (ou seja, os clientes que serão atendidos no dia, suas respectivas demandas e as coordenadas geográficas dos endereços), inseridos na planilha em *Excel* pelo usuário. O programa então realiza automaticamente todas as etapas do algoritmo descrito, a saber: cálculo dos parâmetros iniciais do problema, execução da varredura, resolução dos problemas de caixeiro viajante e mudança do ponto de início da próxima varredura. O código completo elaborado em VBA é apresentado no Anexo B.

### 5.1.2. Rotina de resolução do problema de caixeiro viajante

No modelo a ser desenvolvido, será criada uma rotina para a resolução dos problemas de caixeiro viajante originados após o procedimento de varredura, com a alocação dos pontos de demanda aos veículos, isto é, a formação dos grupos de pontos (*clusters*) a serem atendidos por cada um dos veículos.

Basicamente, esta rotina deve resolver todos os problemas de caixeiro viajante formados em cada iteração do algoritmo de varredura, armazenando os roteiros formados e os custos (distâncias) totais de cada roteiro. A solução da roteirização é a composição de todos os roteiros formados na resolução dos problemas de caixeiro viajante formados durante a varredura.

O problema de caixeiro viajante (TSP – *Traveling Salesman Problem*) é um problema de otimização combinatória muito conhecido e que serve como base para muitos dos

problemas conhecidos de roteirização de veículos. Existem na literatura, diversas formulações de problemas de caixeiro viajante, e, para o modelo desenvolvido no presente trabalho, selecionou-se a formulação de Miller, Tucker e Zemlin (1960), principalmente pela sua facilidade de implantação e programação nas plataformas *Excel* e VBA. A formulação de Miller-Tucker-Zemlin é apresentada no Apêndice A.

A rotina de resolução dos problemas de caixeiro viajante utilizada no modelo proposto, foi criada através do software *Microsoft Excel*, com a criação de um programa que formula o problema completo do caixeiro viajante, (com todas as variáveis e restrições, segundo a formulação de Miller, Tucker e Zemlin) , na forma de planilha. O problema formulado é então resolvido com o auxílio do software de otimização “*What's Best!*”, que funciona integrado à plataforma *Excel*.

## **5.2. VALIDAÇÃO DO MODELO**

A etapa de validação consiste na aplicação do modelo elaborado em problemas reais da empresa, buscando comprovar sua eficiência e precisão, e por consequência, sua utilidade para a empresa em termos práticos. Para tanto, uma série de testes envolvendo problemas de roteirização reais, em casos ocorridos na empresa foram realizados, com o objetivo de testar a robustez do modelo e verificar a sua agilidade computacional em diversas situações.

Para a efetiva validação do modelo proposto, o algoritmo foi testado em uma extensa gama de problemas, agrupados em categorias diferentes, de acordo com as características do tipo de problema simulado. O objetivo de tais simulações é verificar o comportamento do algoritmo em problemas de roteirização com características diferentes (dentro do escopo do modelo), nos seguintes aspectos:

- **Qualidade da solução**

Este parâmetro pode ser definido como a qualidade da solução obtida com a aplicação do algoritmo, em relação à solução real. Dessa forma, a qualidade da solução é mensurada através da comparação da solução obtida heuristicamente com a situação ocorrida efetivamente na empresa, ou seja, os roteiros e o custo total obtidos pelo algoritmo devem ser comparados com os roteiros realizados efetivamente pela empresa, bem como com o custo total destes roteiros.

- **Tempo computacional**

Este parâmetro considera os tempos de computação decorridos para a resolução completa de cada um dos problemas de roteirização testados. O objetivo desta análise é verificar o desempenho computacional do algoritmo nas diversas instâncias de problemas simulados.

- **Proximidade da solução obtida em relação à solução ótima (em instâncias de pequeno porte)**

Nos problemas de pequeno porte, com tamanho máximo em torno de 15 pontos (conforme discutido no item 3.1) a exatidão, ou seja, a proximidade da solução obtida heuristicaamente em relação à solução exata pode ser verificada através da comparação com a solução ótima do problema de roteirização simulado, obtida através do procedimento iterativo descrito no item 3.1.

Os problemas simulados nesta etapa foram extraídos das operações de entrega realizadas no mês de julho de 2007 e enquadrados em diferentes categorias, de acordo com características comuns, visando à formação de um conjunto completo de problemas que representem de modo fiel as operações de entrega de mercadorias realizadas pela empresa estudada. As categorias nas quais os problemas testados foram enquadrados são definidas como segue.

### **Problemas de pequeno porte**

Os problemas enquadrados nesta categoria são caracterizados por serem de pequeno porte, de modo que a solução ótima pode ser obtida rapidamente através da aplicação do algoritmo proposto no item 3.1. Problemas deste tipo são de rara ocorrência nas operações diárias da empresa, mas são de grande utilidade para o estudo de validação e análise de sensibilidade do modelo. Os principais critérios que caracterizam os problemas de pequeno porte foram definidos como:

- Número máximo de pontos de demanda: 15
- Nenhuma demanda individual maior que a capacidade dos veículos
- Total das demandas dos clientes maior que a capacidade de um veículo, forçando a utilização dos dois veículos

## **Problemas de médio porte**

Os problemas enquadrados nesta categoria são de médio porte e cuja solução ótima não pode ser obtida através do método exato em tempos razoáveis de computação. Estes problemas são de ocorrência freqüente na empresa no cotidiano das operações de entrega. Os principais critérios definidos para a caracterização deste tipo de problema são:

- Número de pontos de demanda: entre 20 e 30
- Nenhuma demanda individual maior que a capacidade dos veículos
- Total das demandas dos clientes maior que a capacidade de um veículo, forçando a utilização dos dois veículos

## **Problemas com demandas individuais maiores que a capacidade dos veículos**

Os problemas enquadrados nesta categoria apresentam a característica de um dos pontos de demanda do problema possuir a demanda maior que a capacidade máxima de um veículo, de modo que este não consiga atender toda a demanda deste cliente sozinho. Problemas deste tipo ocorrem esporadicamente no cotidiano da empresa. Os principais critérios que definem os problemas desse tipo são:

- Número de pontos de demanda: entre 10 e 20
- Um dos pontos de demanda maior que a capacidade máxima de um veículo
- Total das demandas dos clientes menor que a soma das capacidades máximas dos dois veículos, de modo que os dois veículos consigam atender todos os pontos com apenas uma viagem cada. Esta restrição será discutida adiante.

Vale ressaltar que neste modelo, a soma das demandas de todos os clientes do dia não pode ultrapassar a soma das capacidades máximas dos veículos disponíveis. (no caso da empresa estudada existem fisicamente apenas dois veículos disponíveis para realizar as entregas). No entanto, ocorrências em que as demandas totais ultrapassam as capacidades máximas totais dos veículos disponíveis são bastante raras no cotidiano das operações de entrega.

Os resultados do estudo de validação do modelo realizado conforme a metodologia apresentada neste capítulo serão apresentados e analisados no capítulo 6.

## 6. ANÁLISE DOS RESULTADOS

Os resultados obtidos nos experimentos realizados na etapa de implantação e validação do modelo são apresentados neste capítulo. Foram realizadas no total 15 simulações de situações reais ocorridas na empresa, divididas de acordo com a classificação estabelecida no capítulo 5, com o intuito de verificar a eficiência do modelo proposto na resolução de problemas de roteirização.

Primeiramente, para efeitos ilustrativos, serão apresentados os resultados de um problema real extraído de um dia cotidiano de entregas da empresa, resolvido através da aplicação do algoritmo proposto, seguidos de uma análise detalhada dos mesmos, em comparação com a situação real. A segunda parte deste capítulo mostra os resultados obtidos nas diversas simulações realizadas na etapa de testes, com uma posterior análise segundo os parâmetros definidos no capítulo 5. Finalmente, com base nas análises realizadas, a última parte discorre acerca dos resultados obtidos, verificando a eficiência do algoritmo proposto, bem como sua aplicabilidade na empresa estudada.

### 6.1. ESTUDO DE CASO

O modelo foi simulado em um problema de roteirização extraído de um dia típico de entregas na empresa estudada. Com base nos dados coletados sobre as operações de entrega, como o histórico de entregas decorridas, observou-se que o exemplo considerado é o que reproduz mais fielmente a rotina das operações de entrega de mercadorias na empresa.

No exemplo considerado, há 24 pontos de demanda a serem atendidos pelos dois veículos de que a empresa dispõe, sendo que cada um deles deve realizar apenas uma única viagem, ou seja, a soma das demandas não ultrapassa a soma das capacidades dos veículos. Os dados deste problema encontram-se no Anexo A deste trabalho. Os resultados obtidos na resolução do problema são apresentados a seguir.

A tabela 6.1.1 mostra os roteiros para os dois veículos obtidos pelo algoritmo em comparação com os roteiros realizados na situação real, bem como a distância total percorrida nos dois casos.

*Tabela 6.1.1 – Resultados obtidos na simulação com 25 pontos*

	<b>Veículo</b>	<b>Rota</b>	<b>Distância da rota</b>	<b>Distância Total</b>
<b>Situação real</b>	1	0-5-6-4-7-3-1-2-0	93,0224	
	2	0-12-18-23-19-9-10-8-11-14-22-16-13-17-15-24-21-20-0	82,7518	<b>135,8239</b>
<b>Solução obtida</b>	1	0-24-16-22-14-11-3-1-2-15-17-13-0	53,3414	
	2	0-5-12-6-4-7-9-10-8-19-18-23-21-20-0	82,4824	<b>175,7742</b>

Neste exemplo, os roteiros obtidos com a aplicação do modelo de roteirização proposto apresentaram uma distância total percorrida de **135,82** km, apresentando uma economia de **40** km, ou **22,7%** em relação aos **175,77** km percorridos nos pelos veículos nos roteiros realizados na situação real. A representação gráfica dos roteiros da situação real e da solução obtida é mostrada no Anexo A. O problema considerado neste item corresponde ao problema de número 14 apresentado na seção seguinte.

O exemplo apresentado mostra as economias potenciais que podem ser obtidas através da aplicação do algoritmo de varredura para a determinação dos roteiros dos veículos. A seção a seguir apresenta uma análise de todas as simulações realizadas, baseadas em situações reais na empresa, como forma de verificar a consistência do modelo em relação à qualidade das soluções obtidas, ou seja, a melhoria obtida pela aplicação do modelo em relação a situação atual.

## 6.2. ANÁLISE DOS RESULTADOS DAS SIMULAÇÕES DO MODELO

O modelo de roteirização proposto foi aplicado em 15 problemas reais encontrados na empresa, enquadrados nas categorias definidas no item 5.2, ou seja, problemas de pequeno porte (com no máximo 15 pontos de demanda), problemas de médio porte (em torno de 20 a 30 pontos) e problemas em que a demanda individual de um cliente supera a capacidade de um veículo. Dessa forma, é possível testar o desempenho do modelo em uma série de problemas com características diferentes que podem ocorrer no cotidiano da empresa.

Os resultados obtidos em todas as simulações do algoritmo são apresentados na tabela 6.2.1 a seguir.

Tabela 6.2.1 – Resultados das simulações do modelo

Problema	Pontos*	Situação real	Solução obtida	Solução exata	Tempo de computação do algoritmo (s)	Classificação
1	10	91,051	84,126	83,265	67	A
2	10	40,933	35,170	35,170	31	A
3	15	109,647	88,823	86,525	157	A
4	15	215,899	177,158	-	228	B
5	15	147,941	110,649	-	121	B
6	15	109,370	76,114	-	249	B
7	15	152,594	129,702	-	166	C
8	20	171,015	134,702	-	454	C
9	20	99,440	79,440	-	242	C
10	21	173,552	153,085	-	465	C
11	21	103,261	87,238	-	352	C
12	22	164,126	135,096	-	434	C
13	23	138,982	98,892	-	484	C
14	24	162,831	135,824	-	600	C
15	25	213,987	186,935	-	961	C

\* Excluindo o depósito

Na tabela 6.2.1, a classificação dos problemas é dada por:

- Classificação A

Enquadra os problemas de pequeno porte, com até 15 pontos de demanda. São de ocorrência mais rara na empresa, em especial os problemas com 10 pontos ou menos.

- Classificação B

Enquadra os problemas que possuem um ponto de demanda maior que a capacidade de um veículo. Em geral, o número médio de pontos atendidos nestas situações é em torno de 15. Tais problemas ocorrem esporadicamente no cotidiano da empresa, em especial nos casos de pedidos de grandes clientes. Vale ressaltar que nos problemas considerados, a soma das demandas de todos os pontos não ultrapassa a soma das capacidades do veículos disponíveis.

- Classificação C

Enquadra os problemas de médio porte (entre 20 e 30 pontos), que são os de ocorrência mais freqüente na empresa.

Os resultados obtidos serão então analisados conforme os parâmetros definidos no item 5.2 do presente trabalho, a saber: qualidade da solução, desempenho computacional e proximidade da solução em relação ao ótimo, no caso dos problemas de pequeno porte.

#### **6.2.1. Qualidade da solução**

Este item consiste de uma análise da qualidade das soluções obtidas nos problemas de roteirização simulados com a aplicação do modelo. No presente trabalho, qualidade da solução é definida como o quanto melhor a solução obtida através do modelo é em relação à situação real encontrada na empresa. Desse modo, é possível verificar as economias que podem ser geradas em termos de distâncias percorridas com a aplicação do modelo, em relação à situação real.

O gráfico da figura 6.2.1 apresenta a comparação entre as soluções obtidas em relação à situação real, em todos os problemas simulados.

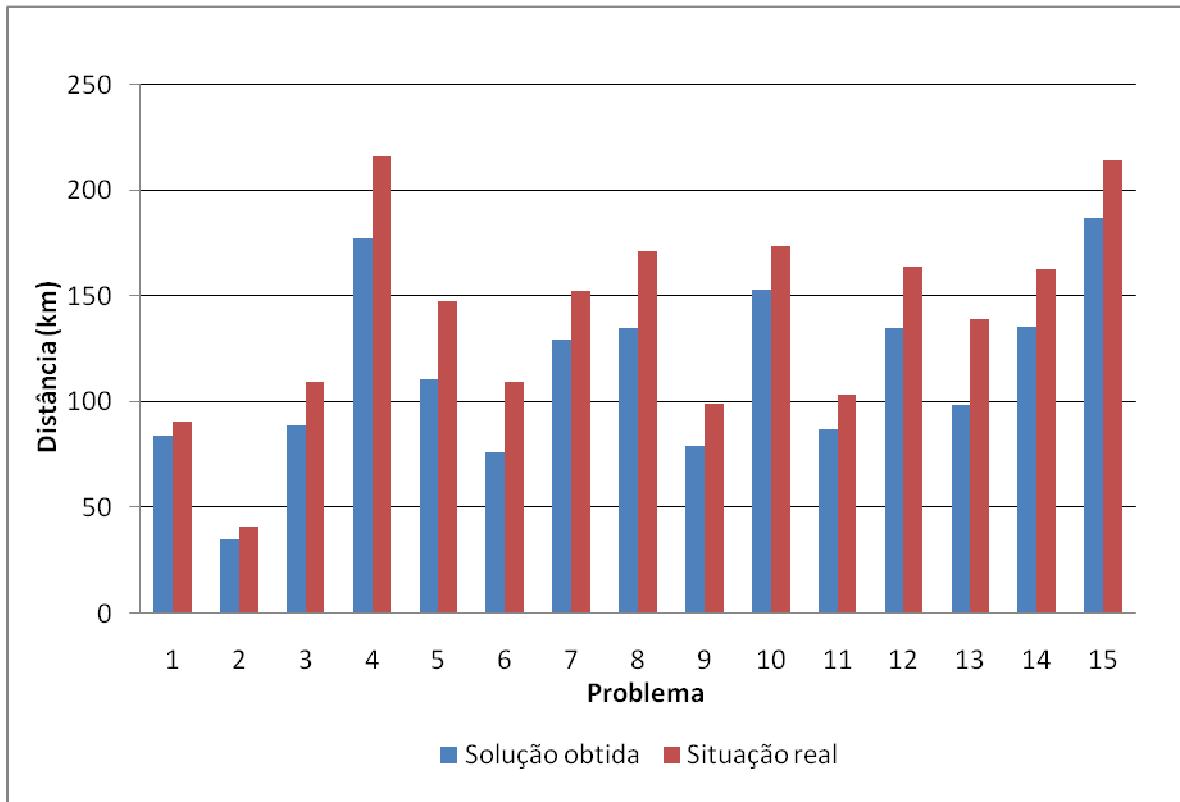


Figura 6.2.1 – Comparação entre as soluções obtidas e a situação real

Observa-se que em todos os problemas simulados que a solução obtida através do algoritmo proposto apresenta uma distância total percorrida pelos veículos menor do que a encontrada na correspondente situação real. A tabela 6.2.2 apresenta a economias obtidas em cada um dos problemas simulados em relação à situação real.

*Tabela 6.2.2 – Economias geradas em relação à situação real*

<b>Problema</b>	<b>Pontos</b>	<b>Situação real (km)</b>	<b>Solução obtida (km)</b>	<b>Economia (km)</b>	<b>Economia (%)</b>
1	10	91,051	84,126	6,925	0,076
2	10	40,933	35,170	5,763	0,141
3	15	109,647	88,823	20,824	0,190
4	15	215,899	177,158	38,740	0,179
5	15	147,941	110,649	37,292	0,252
6	15	109,370	76,114	33,256	0,304
7	15	152,594	129,702	22,892	0,150
8	20	171,015	134,702	36,313	0,212
9	20	99,440	79,440	20,000	0,201
10	21	173,552	153,085	20,466	0,118
11	21	103,261	87,238	16,023	0,155
12	22	164,126	135,096	29,030	0,177
13	23	138,982	98,892	40,090	0,288
14	24	162,831	135,824	27,007	0,166
15	25	213,987	186,935	27,052	0,126

O gráfico da figura 6.2.2 a seguir apresenta as economias obtidas em cada problema simulado em termos absolutos (km).

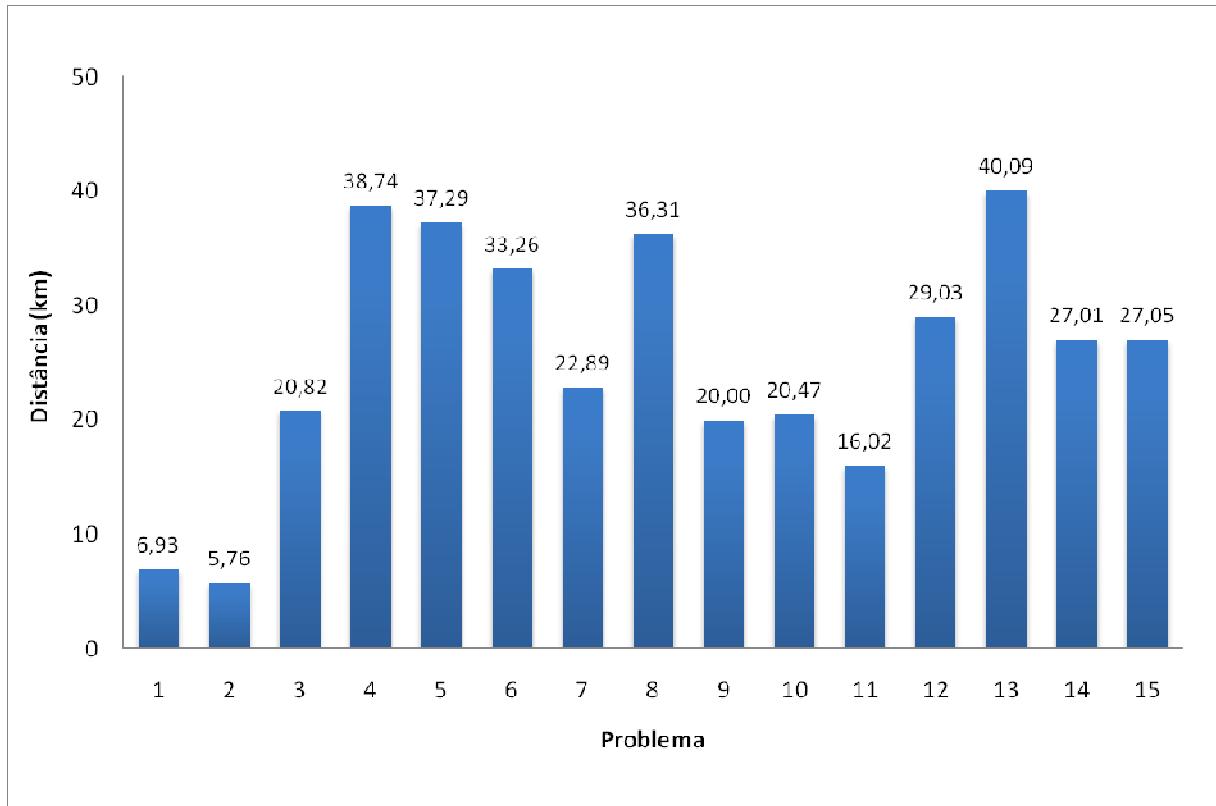


Figura 6.2.2 – Economias geradas com a aplicação do modelo

Através da figura 6.2.2 é possível observar que a aplicação do modelo proposto nos problemas simulados gerou economias significativas, com um valor máximo de **40,09** km no problema 13. O valor médio das economias obtidas nos problemas estudados foi de **25,45** km. Observa-se que as menores economias são referentes aos problemas de menor porte (10 pontos). A figura 6.2.3 mostra as economias obtidas em termos percentuais.

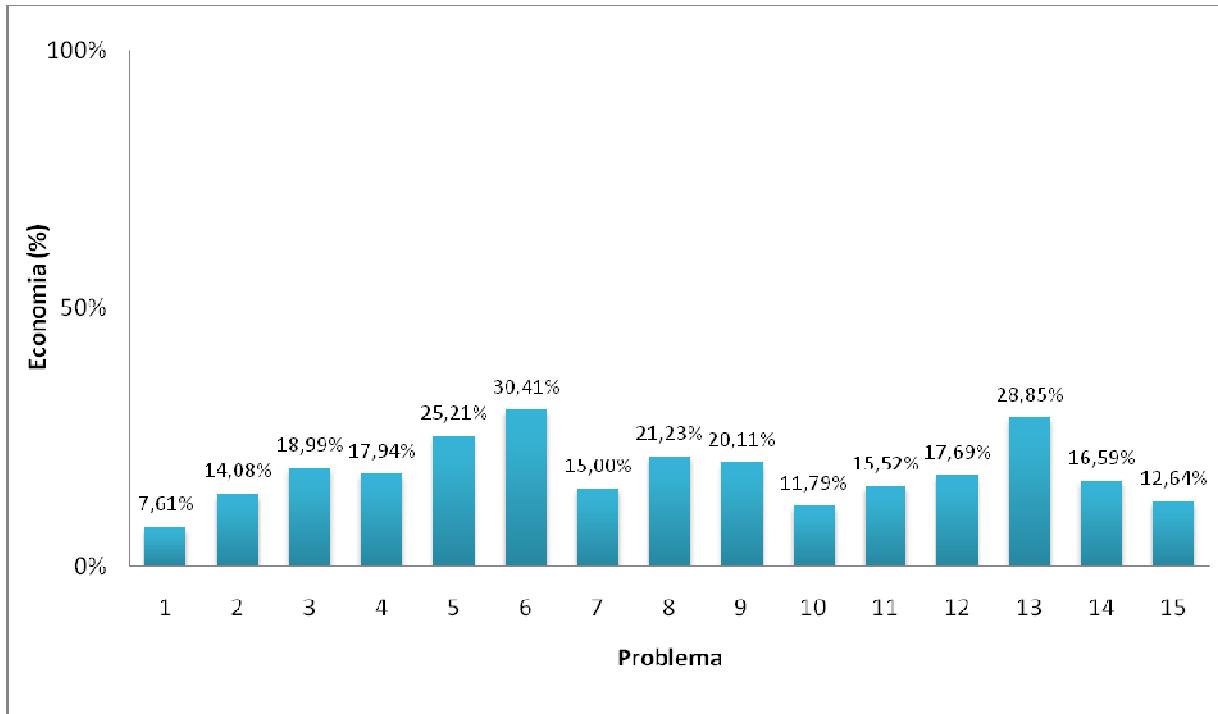


Figura 6.2.3 – Percentual das economias geradas

Pode-se observar na figura 6.2.3 que as soluções obtidas com a aplicação do modelo, trouxeram economias significativas em relação à situação atual, com um percentual médio de **18,24%**. A maior economia obtida foi de **30,46%**, no problema 6, com **33,26** km economizados em termos absolutos. Embora tenha sido o problema com maior economia percentual, em termos absolutos a maior economia foi observada no problema 13, com economia de **40,08** km, ou **28.85%**.

### 6.2.2. Desempenho computacional

Neste item, será apresentada uma análise do desempenho computacional do modelo proposto, em relação aos problemas simulados. O desempenho computacional foi definido como o tempo total de processamento e solução do problema através da aplicação do algoritmo de varredura. O gráfico da figura 6.2.4 apresenta os tempos de computação em segundos, decorridos em cada um dos problemas simulados.

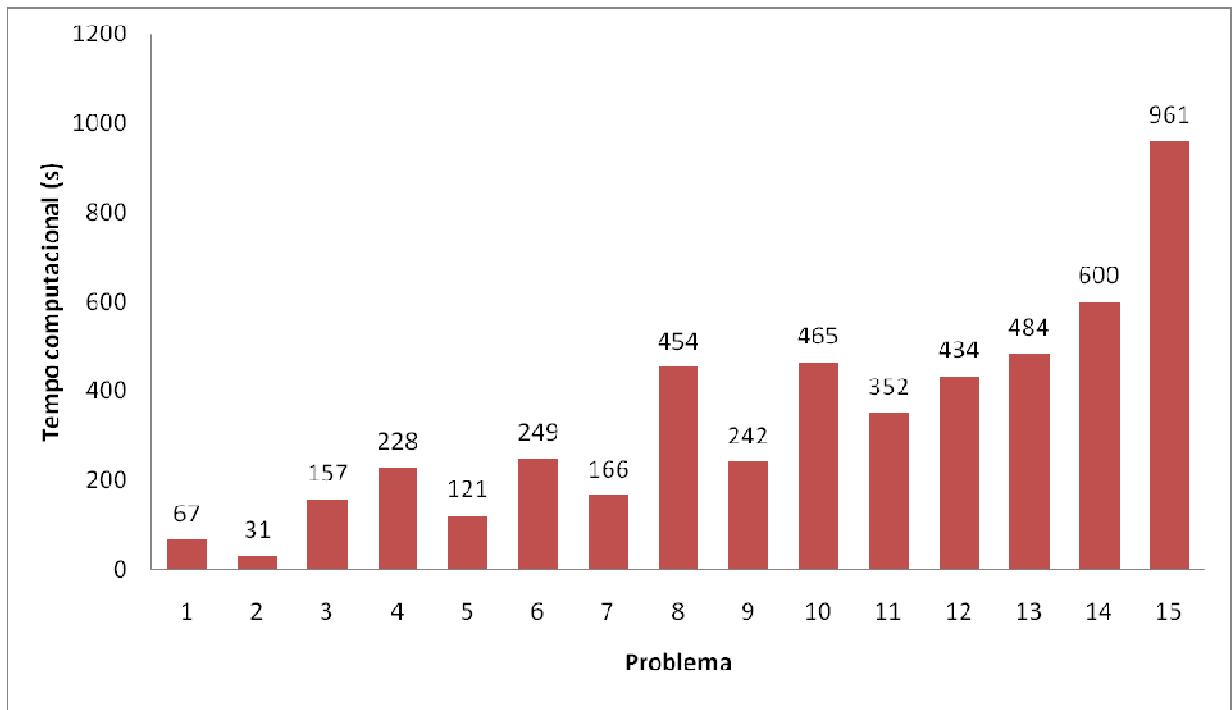


Figura 6.2.4 – Desempenho computacional do algoritmo nos problemas simulados

Pode-se observar pela figura 6.2.4 que os tempos computacionais aumentam conforme o aumento do tamanho do problema. Os primeiros problemas, por serem de tamanho menor (ou seja, com menor número de pontos) apresentam tempos de computação bastante baixos (problemas 1 e 2). Já os problemas maiores, como os de número 14 e 15, apresentam tempos computacionais mais elevados, com um máximo de **961** segundos ou aproximadamente 15 minutos. O tempo médio de computação dos problemas simulados foi de **334,1** segundos.

A figura 6.2.5 apresenta os tempos médios de computação para cada uma das classes de problemas definidas no capítulo 5.

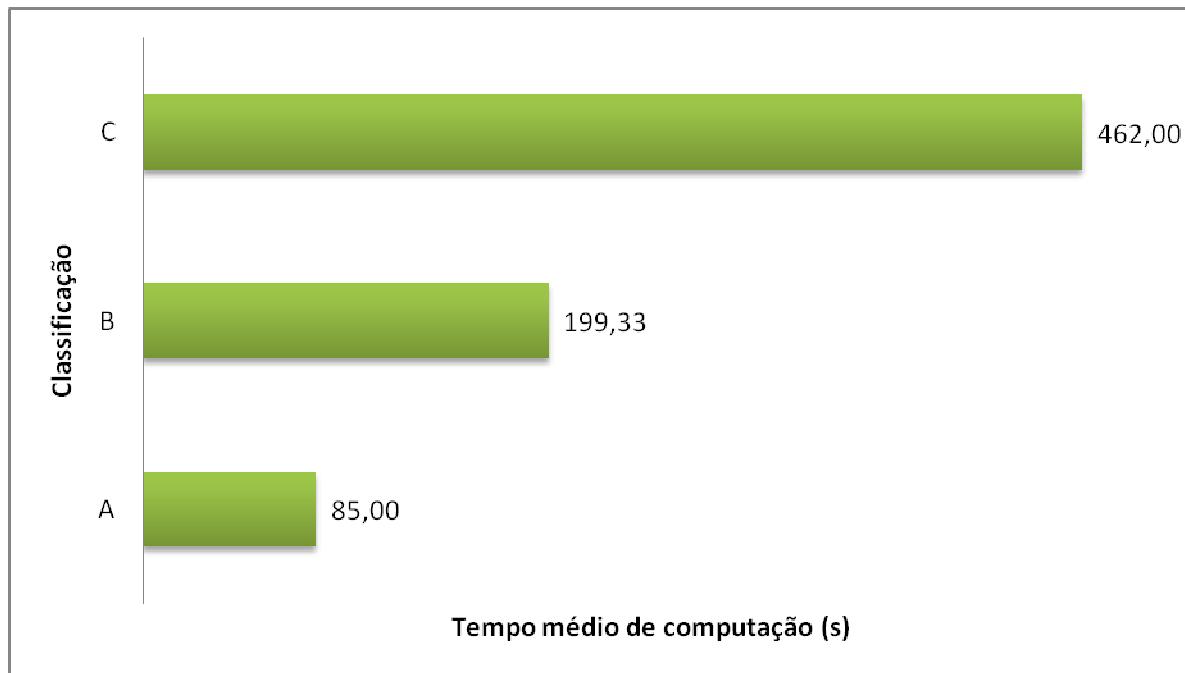


Figura 6.2.5 – Tempo médio de computação por classe de problema (A, B e C)

Observa-se que, devido ao fato de serem problemas de maiores dimensões, os problemas da classe C, ou seja, os problemas de médio porte são os que apresentam o tempo médio de computação mais elevado. A figura 6.2.6 mostra a variação dos tempos computacionais em relação ao número de pontos.

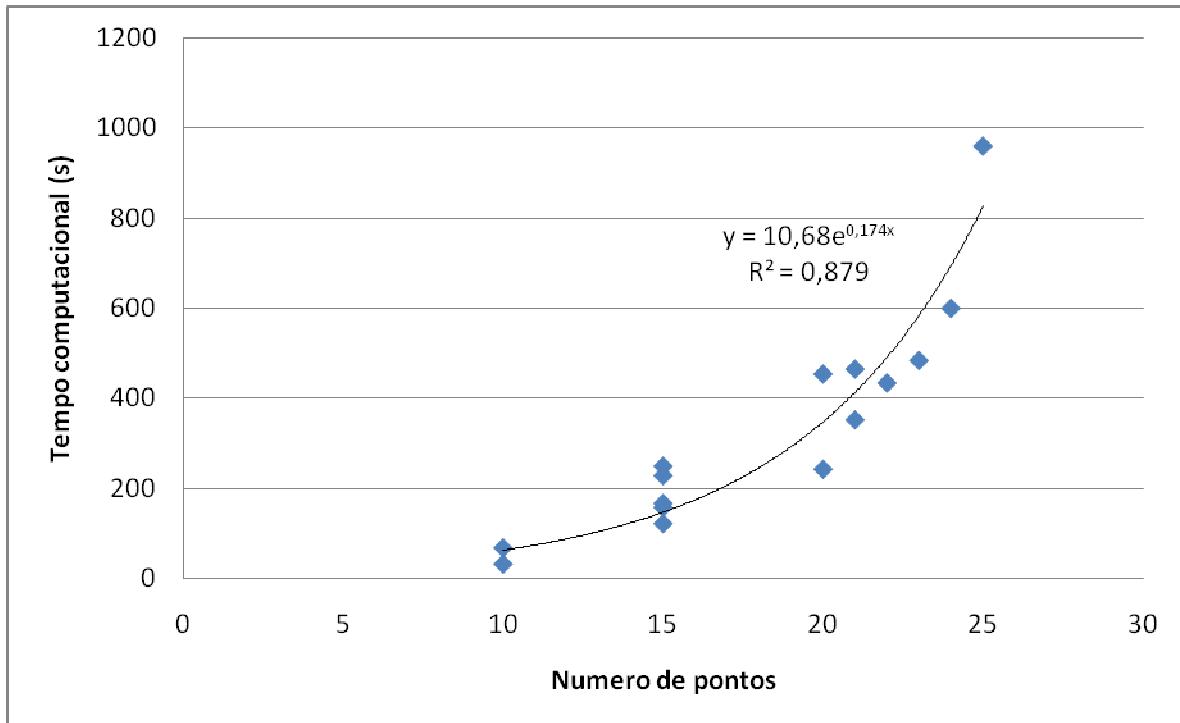


Figura 6.2.6 – Variação dos tempos computacionais com o número de pontos

Observa-se pelo gráfico da figura 6.2.6 que há uma tendência de crescimento exponencial do tamanho do problema à medida que aumenta o tamanho do problema. Basicamente, grande parte dos tempos totais de computação é devida à rotina de resolução dos problemas de caixeiro viajante. Uma vez que, na rotina utilizada no modelo proposto, os problemas são resolvidos através de um método exato, os tempos de resolução aumentam drasticamente conforme aumenta o tamanho dos problemas.

No entanto, no contexto dos problemas de roteirização encontrados na empresa, problemas maiores que os considerados nesta análise são de ocorrência muito rara, de modo que os tempos de computação observados são viáveis para os problemas de roteirização cotidianos na empresa.

### 6.2.3. Exatidão da solução

Nos problemas simulados de pequeno porte, é possível realizar uma comparação entre a solução obtida heuristicamente através da aplicação do algoritmo de varredura e a solução ótima do problema considerado. Nos problemas de pequeno porte simulados, observou-se que as soluções obtidas são bastante próximas do ótimo, sendo que em um dos casos, o modelo retornou a própria solução ótima do problema. A tabela 6.2.3 mostra a comparação entre as

soluções obtidas com a aplicação do modelo e as soluções ótimas obtidas para os problemas de pequeno porte estudados.

*Tabela 6.2.3 – Comparação entre as soluções obtidas com as soluções ótimas*

Problema	Pontos	Solução obtida (km)	Solução exata (km)	Diferença (km)	Diferença (%)
1	10	84,126	83,265	0,860	0,010
2	10	35,170	35,170	0,000	0,000
3	15	88,823	86,525	2,298	0,026

Foi possível observar nos estudos realizados que o modelo fornece soluções bastante próximas às soluções ótimas, sendo que no problema 2 a solução obtida foi a própria solução ótima. A diferença média obtida entre as soluções exata e heurística dos problemas de pequeno porte simulados foi de **1,2%**.

#### 6.2.4. Conclusão dos resultados obtidos

A partir dos resultados obtidos pode-se concluir que o modelo utilizado fornece soluções bastante satisfatórias para os problemas de roteirização encontrados no cotidiano da empresa estudada. Tal fato pode ser comprovado pelas economias significativas, em termos de distâncias percorridas, obtidas pela aplicação do modelo, em relação ao atual processo de roteirização realizado na empresa. A economia média de **25,45 km** ou **18,24%** em relação à situação atual.

Além disso, através de uma análise do desempenho computacional do modelo proposto, foi possível constatar que a dificuldade computacional dos problemas de roteirização apresenta uma tendência de crescimento exponencial conforme o aumento do tamanho dos problemas. No entanto, para as operações rotineiras de distribuição na empresa, o modelo apresentou um desempenho bastante adequado para os problemas da empresa, com tempos computacionais viáveis, que não ultrapassaram a faixa dos 15 minutos.

Finalmente, observou-se nos testes com problemas de pequeno porte, com até 15 pontos de demanda, que o modelo fornece soluções muito próximas às soluções ótimas dos problemas, indicando que as soluções obtidas com aplicação do modelo são viáveis e precisas.

### 6.3. ANÁLISE CRÍTICA

A análise dos resultados obtidos com a utilização do modelo proposto fornece subsídios importantes para a avaliação de sua eficiência, tanto em relação ao desempenho computacional quanto à qualidade da solução obtida. Nesta seção, será então abordada a questão da viabilidade de implantação prática do modelo proposto na empresa estudada.

A partir da análise dos resultados obtidos nas simulações apresentadas no item 6.2 é possível extrair os seguintes pontos.

- O modelo proposto apresenta uma eficiência considerável em termos de qualidade das soluções, uma vez que em todos os experimentos realizados, as economias obtidas em termos da distância total percorrida pelos veículos foram significativas.
- O aumento exponencial observado nos tempos computacionais nos problemas de dimensões maiores se deve principalmente à abordagem exata utilizada na rotina de resolução dos problemas de caixeiro viajante, a qual faz parte do algoritmo principal de resolução. No entanto, estes tempos são viáveis considerando-se as características das operações de entrega da empresa, pois são muito raras as ocorrências de problemas maiores do que os simulados neste trabalho. Além disso, o planejamento das rotas ocorre apenas uma vez por dia de tal maneira que os tempos de computação observados não interfeririam na dinâmica das operações de distribuição.

Pode-se concluir então, com base nos pontos levantados que a implantação do modelo proposto é viável, mostrando os benefícios que podem ser obtidos com a aplicação de um método científico para a determinação dos roteiros a serem percorridos pelos veículos. Estes benefícios se traduzem em uma maior facilidade no processo decisório em relação à operação distribuição das mercadorias e também na geração de economias tanto em termos de distâncias quanto de custos.

Além disso, o modelo traz como vantagem a facilidade de utilização, uma vez que foi desenvolvido no ambiente *Microsoft Excel*, tornando a implantação prática do modelo na empresa estudada ainda mais simples e viável. Em virtude da adaptação do algoritmo original de varredura, o modelo também possibilita o fracionamento das demandas dos clientes, ou seja, permite que um cliente seja atendido por mais de um veículo, ocasionando a geração de

mais economias, em termos de custo e no número de veículos utilizados, além de um melhor aproveitamento das capacidades dos veículos.

## 7. CONCLUSÕES

O presente trabalho de formatura apresentou um modelo heurístico para a resolução de problemas de roteirização de veículos baseados na instância de entregas fracionadas, em uma empresa importadora de produtos japoneses. A proposta deste trabalho atua no sentido de suprir uma necessidade da empresa com relação à atividade de planejamento de roteiros e programação de veículos através da aplicação de uma abordagem científica a um processo que atualmente é realizado de maneira informal e sem rigor nos critérios.

Este trabalho aborda o problema de roteirização de veículos com entregas fracionadas (SDVRP – *Split Delivery Vehicle Routing Problem*), introduzida na literatura por Dror e Trudeau (1990). Nesta instância do problema clássico de roteirização, é permitido o fracionamento da demanda dos clientes a serem atendidos, de modo que um determinado cliente possa ser atendido por mais de um veículo. A possibilidade de fracionamento de entregas mostra-se como uma alternativa bastante interessante em termos de geração de economias, conforme apresentado no decorrer do estudo realizado.

O modelo desenvolvido foi baseado no método heurístico de varredura, apresentado por Gillett e Miller (1974), no artigo “*A Heuristic Algorithm for the Vehicle-Dispatch Problem*”. Inicialmente, o método proposto foi descrito detalhadamente, de modo a ilustrar a sua aplicabilidade no problema considerado. Para tanto, um exemplo ilustrativo foi resolvido passo a passo, mostrando cada etapa de funcionamento do algoritmo. Posteriormente, os dados necessários para a entrada no problema foram levantados, possibilitando a aplicação do modelo em uma série de problemas decorridos na empresa, com o intuito de verificar a sua consistência e robustez.

Os resultados obtidos, nas simulações de problemas reais da empresa, mostraram que, com a aplicação do algoritmo, foi gerada uma economia média de 25,45 km, ou 18,24% na distância percorrida pelos veículos, em relação à situação atual. Além disso, constatou-se que, apesar do crescimento exponencial nos tempos de computação conforme o aumento do tamanho do problema, os tempos obtidos nas simulações não foram excessivos (o tempo máximo foi de aproximadamente 15 minutos), considerando-se as características das operações de entregas da empresa, uma vez que são raras as ocorrências de problemas maiores do que os simulados.

Dessa forma, pode-se concluir que a aplicação de métodos científicos para a resolução dos problemas relacionados ao processo decisório de planejamento de roteiros na empresa estudada é bastante viável de um ponto de vista prático, dados os benefícios alcançados com as economias geradas pela aplicação do modelo proposto

Cabe ressaltar, porém, que o modelo desenvolvido foi simplificado, desconsiderando uma série de restrições e situações que podem ocorrer na realidade e que ampliariam a possibilidade de sua aplicação a problemas com características diferentes dos encontrados na empresa estudada. Dentre eles, pode-se citar: tempos de viagem (tempos de ciclo), restrições de jornada de trabalho, janelas de tempo de atendimento aos clientes, realização de múltiplas viagens, além de considerar somente a distância como critério de decisão, em detrimento a uma estrutura de custos mais adequada.

Outro aspecto que poderia ser considerado em estudos futuros é a realização de uma etapa de melhoria da solução obtida com a aplicação do algoritmo de varredura, conforme apresentado por Gillett e Miller (1974). A melhoria é obtida através de uma busca local, verificando a possibilidade de troca de pontos de demanda entre os roteiros formados, com o objetivo de reduzir ainda mais o custo final da solução.

## REFERÊNCIAS BIBLIOGRÁFICAS

- ARCHETTI, C.; SAVELSBERGH, M.W.P.; SPERANZA, M.G. **An Optimization-Based Heuristic for the Split Delivery Vehicle Routing Problem**, Technical Report n.264, Department of Quantitative Methods, University of Brescia. Italy, 2006.
- BALLOU, R. H. **Business Logistics Management**. Englewood Cliffs NJ: Prentice Hall, 5<sup>a</sup> Ed., 2005.
- BELFIORE, P. P. *Scatter Search para Problemas de Roteirização de Veículos com Frota Heterogênea, Janelas de Tempo e Entregas Fracionadas*. Tese de Doutorado – Escola Politécnica, Universidade de São Paulo. São Paulo, 2006.
- CHEN, S.; GOLDEN, B.; WASIL, E. The Split Delivery Vehicle Routing Problem: Applications, Algorithms, Test Problems and Computational Results. In: EURO XXI, Reykjavik, Iceland, 2006. *Anais...* Reykjavik: EURO XXI.
- CHRISTOFIDES, N. Vehicle Routing. In: **The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization**, LAWLER, E.L.; LENSTRA, J.K.; RINNOOY KAN, A.H.G.; SHMOYS, D.B. (Eds), John Wiley & Sons, 1985.
- CUNHA, ALEXANDRE B. What is behind the Brazilian stabilization?. **Revista Brasileira de Economia**, v.55, n.1, p.133-146, 2001.
- DROR, M.; LAPORTE, G. TRUDEAU, P. Vehicle Routing with Split Deliveries. **Discrete Applied Mathematics**, v.50, n.3, p.239-254, 1994.
- DROR, M.; TRUDEAU, P. Savings by Split Delivery Routing. **Transportation Science**, v.23, n.2, p.141-145, 1989.
- DROR, M.; TRUDEAU, P. Split Delivery Routing. **Naval Research Logistics**, v.37, n.3, p.383-402, 1990.
- FISHER, M.; JAIKUMAR, R. A Generalized Assignment Heuristic for Vehicle Routing. **Networks**, v.11, p.113-129, 1981.
- GAREY, M.R.; JOHNSON, D.S. **Computers and Intractability: A Guide to the Theory of NP-completeness**, W.H.Freeman, 1979.

GILLETT, B.L.; MILLER, L. A Heuristic Algorithm for the Vehicle-Dispatch Problem. **Operations Research**, v.22, n.4, p.340-349, 1974.

GOLDBARG, M.C.; LUNA, H.P. **Otimização Combinatória e Programação Linear – Modelos e Algoritmos**. Rio de Janeiro, Campus, 2000.

LAPORTE, G.; GENDREAU, M.; POTVIN, J.; SEMET, F. Classical and Modern Heuristics for the Vehicle Routing Problem. **International Transactions in Operational Research**. v.7, p.285-300, 2000.

LENSTRA, J.K.; RINNOOY KAN, A.H.G. Complexity of Vehicle and Scheduling Problems. **Networks**, v11, n.2, p.221-227, 1981.

MILLER, C.E.; TUCKER, A.W.; ZEMLIN, R.A. Integer Programming Formulations and Traveling Salesman Problems, **Journal of the Association for Computing Machinery**, v.7, n.4, p.326-329, 1960.

MIURA, M. **Resolução do Problema de Roteamento de Veículos em uma Empresa Transportadora**. Trabalho de Formatura – Escola Politécnica, Universidade de São Paulo. São Paulo, 2003.

PATAKI, G. Teaching Integer Programming Formulations Using the Traveling Salesman Problem. **SIAM Review**, v.45, n.1, p. 116-123, 2003

SCHUSTER, E.W.; ALLEN, S.J. Raw Material Management at Welch's, Inc. **Interfaces**, v.28, p.13-24, 1998.

VIANNA, N.A. **Resolução de um Problema de Transporte Escolar Utilizando Métodos de Otimização**. Trabalho de Formatura, Escola Politécnica, Universidade de São Paulo. São Paulo, 2002.

## APÊNDICE A – Formulação do problema de caixeiro viajante

O problema de caixeiro viajante (*TSP – Traveling Salesman Problem*) é um problema de otimização combinatória muito conhecido na literatura, principalmente devido à sua complexidade e dificuldade de resolução. De fato, os problemas de caixeiro viajante são enquadrados na categoria de complexidade *NP-hard*, de modo que o tempo de resolução cresce exponencialmente com o aumento do tamanho do problema (LENSTRA e RINNOOY KAN, 1985).

O problema de caixeiro viajante deriva de trabalhos atribuídos a William Rowan Hamilton, que propôs um jogo cujo objetivo era encontrar uma rota passando por todos os vértices de um dodecaedro, sem passar duas vezes pelo mesmo ponto. Dessa forma, define-se caminho hamiltoniano como um caminho em um grafo não direcionado, no qual cada vértice é visitado exatamente uma única vez.

Sendo assim, o objetivo do problema do caixeiro viajante consiste em determinar o ciclo hamiltoniano de menor custo em um grafo  $G(N,A)$ , com custos  $c_{ij}$  associados aos arcos.

Em outras palavras, o problema do caixeiro viajante consiste em determinar o roteiro de menor custo, passando por todos os vértices de um grafo uma única vez e retornando ao ponto inicial.

A formulação mais conhecida do problema do caixeiro viajante é de autoria de Dantzig, Fulkerson e Johnson (1954), utilizando variáveis binárias e é apresentada a seguir.

$$\min \sum_{i=1}^n \sum_{j=1}^n c_{ij} x_{ij} \quad (\text{A-1})$$

s.a.

$$\sum_{i=1}^n x_{ij} = 1 \quad i = 1, \dots, n; \quad (\text{A-2})$$

$$\sum_{j=1}^n x_{ij} = 1 \quad j = 1, \dots, n \quad (\text{A-3})$$

$$x_{ij} \in \{0,1\} \quad (\text{A-4})$$

Onde a variável  $x_{ij}$  pode assumir os valores:

$$x_{ij}^v = \begin{cases} 1, & \text{se o arco } (i, j) \text{ faz parte da solução} \\ 0, & \text{caso contrário} \end{cases}$$

O objetivo do problema (A-1) é minimizar o custo (ou distância) total do percurso. As restrições (A-2) e (A-3) garantem que cada vértice  $j$  possua apenas um arco de entrada e um arco de saída, assegurando que cada ponto seja visitado uma única vez. No entanto, estas restrições não são suficientes para impedir a formação de subrotas (*subtours*) desconexas do ponto inicial.

Nesta formulação, as subrotas são eliminadas através da aplicação do conjunto de restrições (A-5).

$$\sum_{i, j \in S} x_{ij} \leq |S| - 1 \quad S \subseteq N \setminus \{1\}; \quad 2 \leq |S| \leq n - 1 \quad (\text{A-5})$$

Onde  $S$  é o conjunto de subgrafos de  $G(N,A)$ , o grafo principal do problema, excluindo-se o ponto inicial. Nota-se que um subgrafo de  $G$  é qualquer grafo formado por pontos de  $G$  com no mínimo 2 pontos e no máximo  $n-1$  pontos. O número de pontos de um subgrafo é denotado por  $|S|$ .

Fazendo com que a quantidade de arcos de um subgrafo seja menor ou igual ao número de vértices deste subgrafo, não haverá a possibilidade destes arcos formarem um circuito fechado neste subgrafo, impedindo então a formação de subrotas.

A grande desvantagem desta formulação é o elevado número de restrições de subrotas, que cresce exponencialmente com o tamanho do problema, tornando-o inviável de ser resolvido em problemas com um grande número de pontos.

### **Formulação de Miller, Tucker e Zemlin (1960)**

Miller, Tucker e Zemlin (1960), criaram um conjunto de restrições e variáveis adicionais para a eliminação de subrotas, alternativamente às restrições formuladas por Dantzig, Fulkerson e Johnson (1954). Na formulação de Miller, Tucker e Zemlin, o conjunto de restrições (A-5) é substituído por:

$$u_1 = 1 \quad (\text{A-6})$$

$$u_i - u_j + 1 \leq (n-1)(1-x_{ij}) \quad \forall i \neq 1 \quad (\text{A-7})$$

$$2 \leq u_i \leq n \quad \forall i \neq 1; \forall j \neq 1 \quad (\text{A-8})$$

Onde a variável  $u_i$  é definida como a seqüência na qual o ponto  $i$  é visitado no decorrer do percurso.

As subrotas são eliminadas, pois:

- (i) A restrição de arcos (A-8) obriga que  $u_j \geq u_i + 1$ , quando  $x_{ij} = 1$ , ou seja, faz com que o ponto  $j$  somente seja visitado depois do ponto  $i$ , estabelecendo assim o seqüenciamento dos pontos a serem visitados.
- (ii) Caso houvesse a formação de duas rotas em uma solução do problema, uma delas não iria conter o ponto inicial (nó 1) e consequentemente não seria possível uma atribuição factível das variáveis  $u_i$ . Como o seqüenciamento é realizado a partir do ponto inicial, com  $u_1 = 1$ , não seria possível estabelecer valores para as variáveis  $u_i$  e estes iriam crescer para infinito (PATAKI, 2003), violando a restrição (A-8).

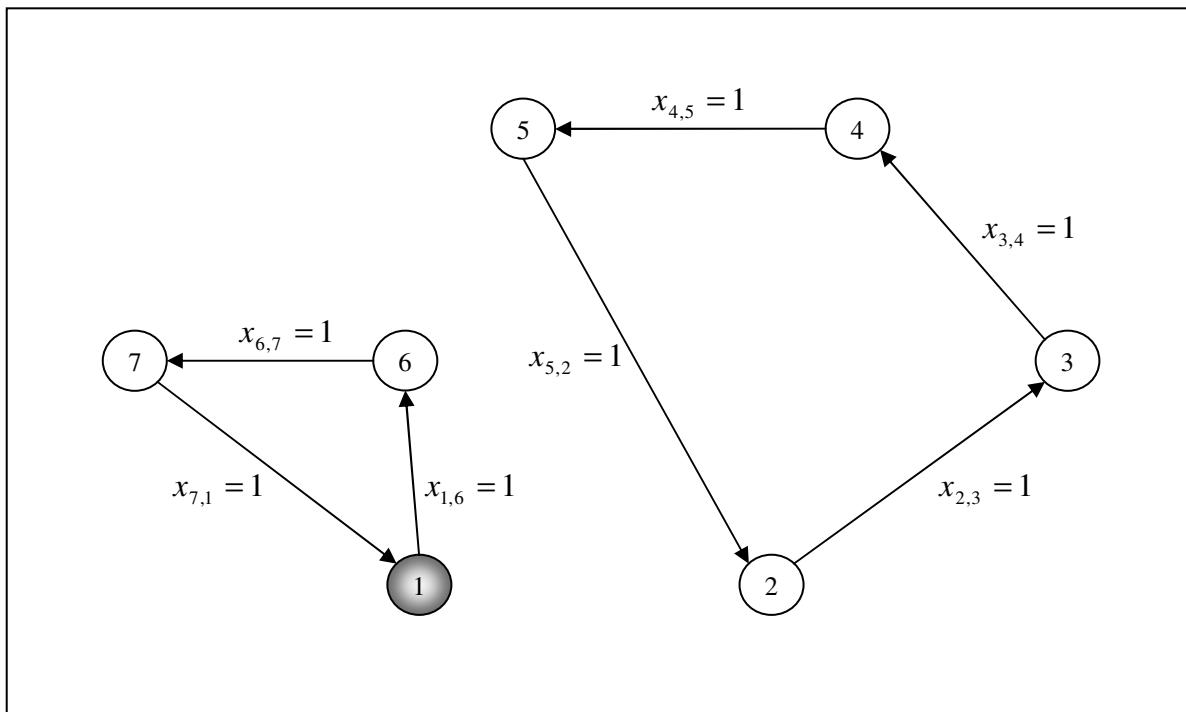


Figura 1 – Eliminação de subrotas na formulação de Miller, Tucker e Zemlin

No exemplo da figura 1, há duas rotas, sendo que uma delas (a rota 2-3-4-5-2) é desconexa do ponto inicial (nó 1). Observa-se neste exemplo que:  $u_1 = 1$ ;  $u_6 = 2$  e  $u_7 = 3$ , mas não é possível determinar os valores para as variáveis  $u_2$ ,  $u_3$ ,  $u_4$  e  $u_5$ , tornando a situação mostrada impossível.

A formulação de Miller, Tucker e Zemlin apresenta uma série de vantagens, tais como:

- Apresenta um número relativamente pequeno de restrições de eliminação de subrotas, com a necessidade de inclusão de  $(n-1)$  variáveis e cerca de  $n^2/2$  restrições adicionais.
- Além disso, caso um ponto  $i$  tenha prioridade de atendimento, um termo  $k \cdot u_i$  pode ser adicionado ao objetivo, forçando a variável  $u_i$  a receber um valor pequeno. As variáveis do tipo  $u_i$  também podem ser ajustadas para comportar restrições de janelas de tempo e capacidade dos caixeiros (PATAKI, 2003).

## **ANEXO A – DADOS DO EXEMPLO DO ESTUDO DE CASO**

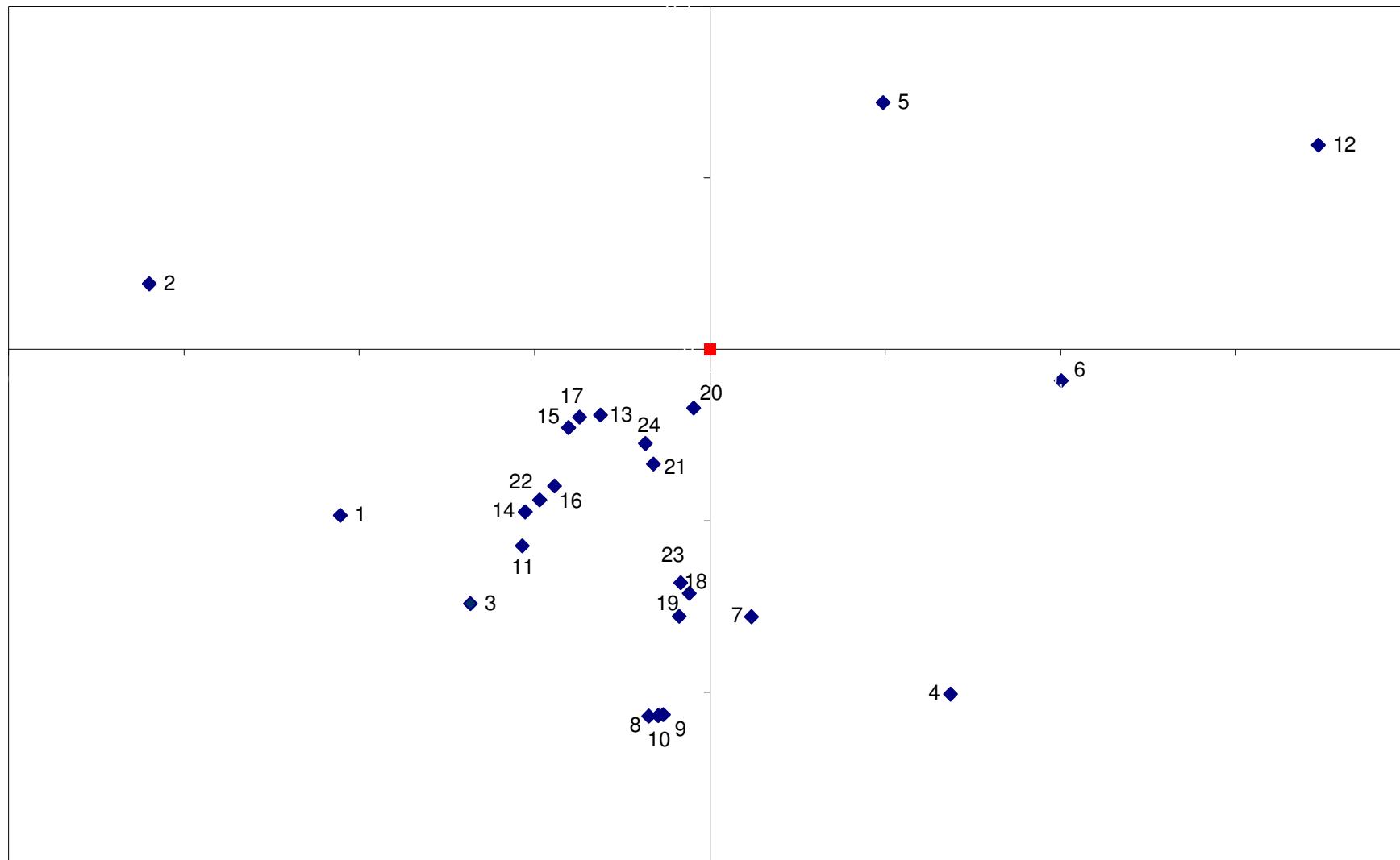
No problema considerado no estudo de caso apresentado no capítulo 6, há 24 pontos de demanda a serem atendidos pelos veículos. Os dois veículos disponíveis na empresa realizam as entregas em apenas uma viagem, ou seja, não há retornos ao depósito para o carregamento de novas mercadorias. Os dados do problema considerado foram extraídos da operação de entrega do dia 05/07/2007 e são mostrados a seguir.

**A-1) Dados dos clientes atendidos no dia considerado**

Ponto	Cliente	Endereço	Bairro	Cidade
1	SUPERMERCADO BERGAMINI LTDA	AV LUIZ STANATIS, 431	JACANA	SAO PAULO
2	CIA. BRASILEIRA DE DISTRIBUICAO	ROD. ANHANGUERA, KM 17.800 - BL. B		OSASCO
3	PONKA COM. PROD. JAPONESES LTDA-ME	AV. MINISTRO LAUDO FERR. DE CAMARGO, 108	JARDIM PERI-PERI	SAO PAULO
4	VIA COSTEIRA COM. DE ART. P/PESCA CAMPI	RUA GALVAO BUENO, 218 - ST.1/2 13/14	LIBERDADE	SAO PAULO
5	MORUMBY HOTEIS LTDA	RUA SAMPAIO VIANA, 294	PARAISO	SAO PAULO
6	MERCEARIA NIHON-HIN LTDA	RUA AUGUSTA, 1961	CONSOLACAO	SAO PAULO
7	A. IWAMA MERCEARIA - ME	RUA LOPES NETO, 308	ITAIM BIBI	SAO PAULO
8	ASSOCIACAO KARAOKE SÃO CAETANO DO SUL	RUA PADRE ANCHIETA, 112	BOA VISTA	SAO CAETANO DO SUL
9	COMERCIAL SAWAKI LTDA	AV.DAS NACOES UNIDAS, 13301	V.CORDEIRO	SAO PAULO
10	MINE'S CONVENIENCIA LTDA - ME	RUA DA CONSOLACAO, 3610	CERQUEIRA CESAR	SAO PAULO
11	GATE GOURMET LTDA	RUA MARIA GARCIA UTRILLA, 453	CUMBICA	GUARULHOS
12	TEMAKARIA NOYOI COM. DE ALIMENTACAO E BEBIDAS LTDA	RUA DR.RENATO PAES BARROS, 65	ITAIM BIBI	SAO PAULO
13	ANIKI PRES. E UTIL. DOM. LTDA-ME.	RUA MANOEL GUEDES, 436	ITAIM BIBI	SAO PAULO

14	JAM WAREHOUSE COM. DE ALIM. E PROMO. LTDA	RUA GOMES CARVALHO, 1152	VILA OLIMPIA	SAO PAULO
15	FM RESTAURANTE LTDA-EPP	RUA DA CONSOLACAO, 3147	CERQUEIRA CESAR	SAO PAULO
16	CHOPERIA RESTAURANTE GREY LTDA	RUA JUNU, 125	VILA CARRAO	SAO PAULO
17	EDO COMERCIO DE ALIMENTOS LTDA	RUA DAS ROSAS,139	PRACA DA ARVORE	SAO PAULO
18	SANDAI COMESTIVEIS LTDA	AV. BOSQUE DA SAUDE, 173	SAUDE	SAO PAULO
19	MERCEARIA AOZORA LTDA - ME	RUA DAS ANINGAS, 348	JABAQUARA	SAO PAULO
20	KAISEN PRODUTOS ALIMENTICIOS LTDA	AVENIDA PAULISTA, 525 LOJA-9	BELA VISTA	SAO PAULO
21	ENILVA DIAS HATA RESTAURANTE -ME	AV JABAQUARA, 1588	SAUDE	SAO PAULO
22	RAM - COMERCIO DE ALIMENTOS E BEBIDAS LTDA	AV. DO CURSINO, 1788	JARDIM DA SAUDE	SAO PAULO
23	DAI-KIM COM. PROD. ALIM. LTDA - ME	RUA BURITIS, 264	JABAQUARA	SAO PAULO
24	ELLYE KIYOMI ISHIY - ME	AV. ENGENHEIRO GEORGE CORBISIER, 1177	JABAQUARA	SAO PAULO

**A-2) Disposição geográfica dos pontos do problema**



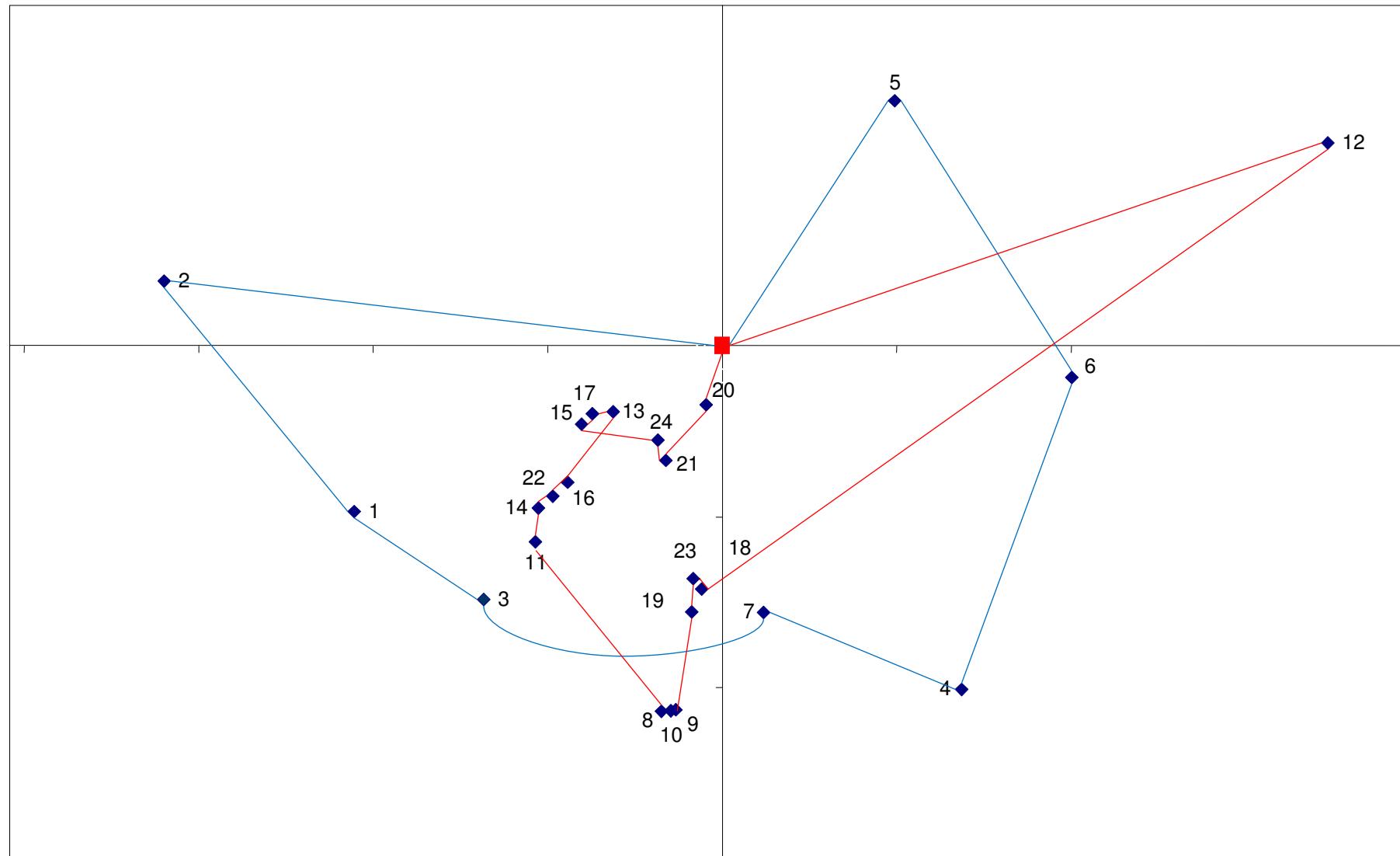
### A-3) Demandas e coordenadas dos clientes

Ponto	Cliente	Veículo	Demandra	LAT	LONG
1	SUPERMERCADO BERGAMINI LTDA	1	8	-23,58882	-46,736
2	CIA. BRASILEIRA DE DISTRIBUICAO	1	34	-23,52121	-46,79
3	PONKA COM. PROD. JAPONESES LTDA-ME	1	2	-23,6146	-46,699
4	VIA COSTEIRA COM. DE ART. P/ PESCA CAMPI	1	2	-23,64092	-46,562
5	MORUMBY HOTEIS LTDA	1	7	-23,46823	-46,581
6	MERCEARIA NIHON-HIN LTDA	1	2	-23,5494	-46,53
7	A. IWAMA MERCEARIA - ME	1	5	-23,61836	-46,619
8	ASSOCIACAO KARAOKE SÃO CAETANO DO SUL	2	2	-23,64733	-46,648
9	COMERCIAL SAWAKI LTDA	2	10	-23,64691	-46,644
10	MINE'S CONVENIENCIA LTDA - ME	2	1	-23,64713	-46,645
11	GATE GOURMET LTDA	2	1	-23,59762	-46,684
12	TEMAKARIA NOYOI COM. DE ALIMENTACAO E BEBIDAS LTDA	2	2	-23,4807	-46,457
13	ANIKI PRES. E UTIL. DOM. LTDA-ME.	2	3	-23,55954	-46,662
14	JAM WAREHOUSE COM. DE ALIM. E PROMO. LTDA	2	3	-23,58777	-46,683
15	FM RESTAURANTE LTDA-EPP	2	2	-23,5631	-46,671
16	CHOPERIA RESTAURANTE GREY LTDA	2	1	-23,58016	-46,675
17	EDO COMERCIO DE ALIMENTOS LTDA	2	1	-23,56005	-46,668
18	SANDAI COMESTIVEIS LTDA	2	4	-23,61151	-46,636
19	MERCEARIA AOZORA LTDA - ME	2	2	-23,61817	-46,639
20	KAISEN PRODUTOS ALIMENTICIOS LTDA	2	4	-23,55749	-46,635
21	ENILVA DIAS HATA RESTAURANTE -ME	2	3	-23,5738	-46,647
22	RAM - COMERCIO DE ALIMENTOS E BEBIDAS LTDA	2	1	-23,58419	-46,679
23	DAI-KIM COM. PROD. ALIM. LTDA - ME	2	14	-23,6084	-46,639
24	ELLYE KIYOMI ISHIY - ME	2	2	-23,56775	-46,649

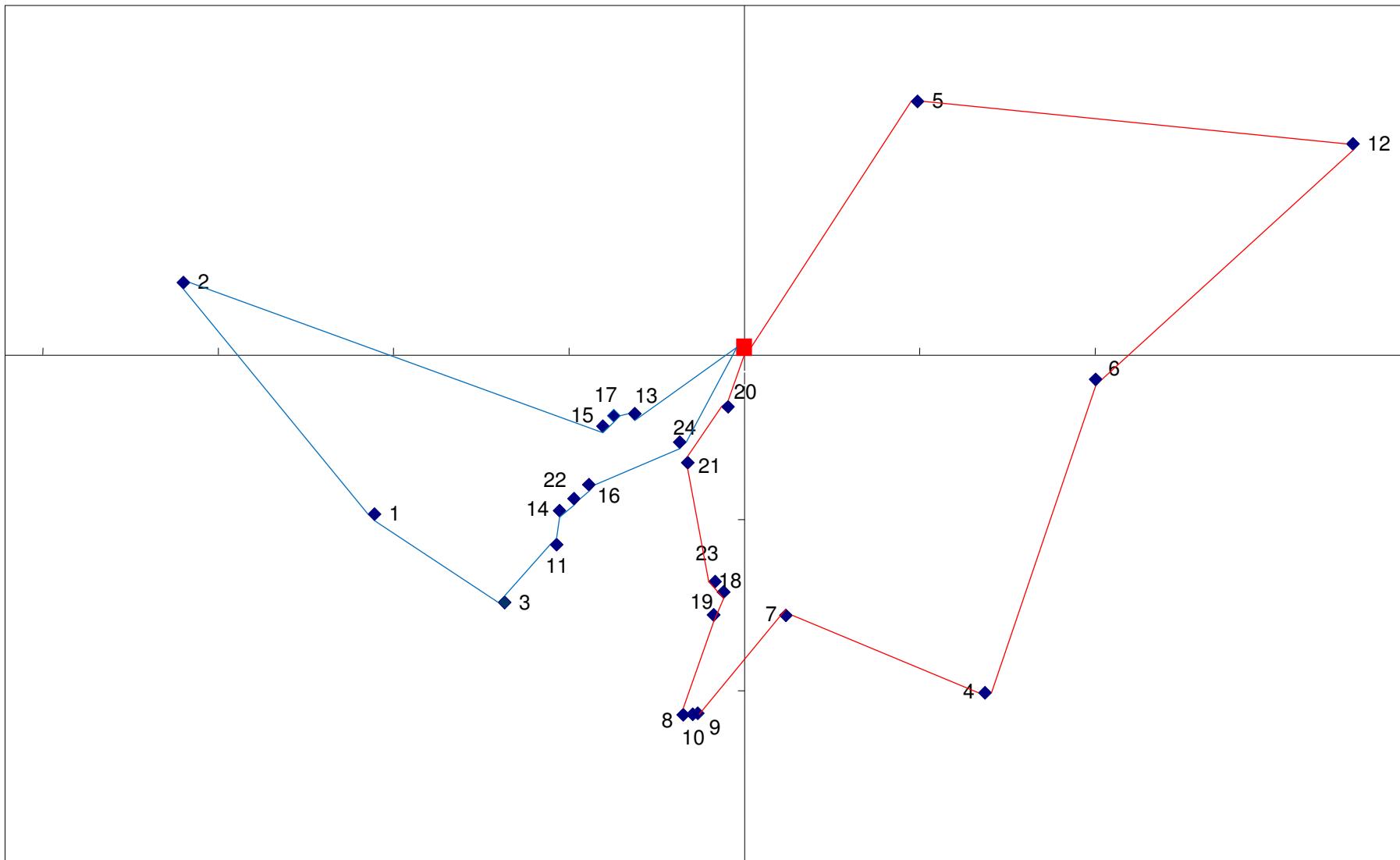
#### A-4) Matriz de distâncias do problema

De (i)	Para (j)																								
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
0	0	14,80	20,24	13,30	16,24	11,66	12,63	10,79	14,81	14,69	14,74	10,33	23,26	4,72	9,27	5,95	7,78	5,39	9,77	10,72	2,42	5,01	8,56	9,38	4,41
1	14,80	0	11,50	5,85	22,97	25,50	26,36	15,26	13,65	14,03	13,90	6,62	38,01	10,14	6,62	8,90	7,77	9,43	12,87	12,78	13,36	11,39	7,16	12,47	11,29
2	20,24	11,50	0	17,19	33,03	27,26	32,87	25,32	24,86	25,19	25,08	16,96	42,22	16,98	16,25	16,07	16,61	16,30	22,94	23,15	20,11	19,43	16,41	22,45	18,86
3	13,30	5,85	17,19	0	17,55	24,89	22,94	10,06	7,80	8,19	8,06	2,97	35,45	8,86	4,16	7,87	5,60	8,43	7,83	7,49	11,18	8,60	4,84	7,56	8,96
4	16,24	22,97	33,03	17,55	0	23,76	13,14	7,77	10,82	10,31	10,49	16,43	25,58	16,77	16,87	17,33	16,42	17,28	10,18	10,19	14,66	14,06	16,63	10,64	14,82
5	11,66	25,50	27,26	24,89	23,76	0	12,81	21,09	25,91	25,70	25,78	21,92	15,67	16,09	20,79	17,19	19,31	16,61	20,81	21,79	13,98	16,63	20,08	20,51	16,07
6	12,63	26,36	32,87	22,94	13,14	12,81	0	14,56	19,93	19,51	19,66	20,38	13,16	16,55	19,89	17,73	18,60	17,29	15,79	16,59	13,20	14,97	19,26	15,83	15,09
7	10,79	15,26	25,32	10,06	7,77	21,09	14,56	0	5,40	5,02	5,16	8,68	27,69	9,70	9,12	10,00	8,76	10,07	2,42	2,58	8,59	7,04	8,91	2,88	7,90
8	14,81	13,65	24,86	7,80	10,82	25,91	19,93	5,40	0	0,51	0,33	8,18	33,07	12,14	9,28	11,89	9,79	12,20	5,11	4,14	12,40	10,07	9,49	5,45	10,90
9	14,69	14,03	25,19	8,19	10,31	25,70	19,51	5,02	0,51	0	0,18	8,42	32,67	12,17	9,48	11,96	9,93	12,26	4,93	3,98	12,29	10,01	9,66	5,31	10,86
10	14,74	13,90	25,08	8,06	10,49	25,78	19,66	5,16	0,33	0,18	0	8,34	32,82	12,17	9,42	11,94	9,89	12,25	5,00	4,03	12,34	10,04	9,61	5,36	10,88
11	10,33	6,62	16,96	2,97	16,43	21,92	20,38	8,68	8,18	8,42	8,34	0	32,68	5,92	1,35	5,01	2,66	5,54	6,27	6,29	8,24	5,72	1,94	5,85	6,01
12	23,26	38,01	42,22	35,45	25,58	15,67	13,16	27,69	33,07	32,67	32,82	32,68	0	27,87	31,95	29,11	30,53	28,58	28,77	29,62	24,71	27,00	31,27	28,75	26,88
13	4,72	10,14	16,98	8,86	16,77	16,09	16,55	9,70	12,14	12,17	12,17	5,92	27,87	0	4,71	1,24	3,26	0,74	7,79	8,51	3,35	2,72	4,02	7,28	1,96
14	9,27	6,62	16,25	4,16	16,87	20,79	19,89	9,12	9,28	9,48	9,42	1,35	31,95	4,71	0	3,72	1,49	4,27	6,71	6,91	7,32	4,97	0,71	6,24	5,10
15	5,95	8,90	16,07	7,87	17,33	17,19	17,73	10,00	11,89	11,96	11,94	5,01	29,11	1,24	3,72	0	2,39	0,58	7,91	8,52	4,54	3,37	3,07	7,38	2,82
16	7,78	7,77	16,61	5,60	16,42	19,31	18,60	8,76	9,79	9,93	9,89	2,66	30,53	3,26	1,49	2,39	0	2,89	6,44	6,85	5,86	3,63	0,78	5,93	3,65
17	5,39	9,43	16,30	8,43	17,28	16,61	17,29	10,07	12,20	12,26	12,25	5,54	28,58	0,74	4,27	0,58	2,89	0	8,06	8,72	4,09	3,24	3,60	7,54	2,57
18	9,77	12,87	22,94	7,83	10,18	20,81	15,79	2,42	5,11	4,93	5,00	6,27	28,77	7,79	6,71	7,91	6,44	8,06	0	0,98	7,40	5,32	6,53	0,53	6,19
19	10,72	12,78	23,15	7,49	10,19	21,79	16,59	2,58	4,14	3,98	4,03	6,29	29,62	8,51	6,91	8,52	6,85	8,72	0,98	0	8,32	6,15	6,83	1,34	7,01
20	2,42	13,36	20,11	11,18	14,66	13,98	13,20	8,59	12,40	12,29	12,34	8,24	24,71	3,35	7,32	4,54	5,86	4,09	7,40	8,32	0	2,66	6,62	6,99	2,23
21	5,01	11,39	19,43	8,60	14,06	16,63	14,97	7,04	10,07	10,01	10,04	5,72	27,00	2,72	4,97	3,37	3,63	3,24	5,32	6,15	2,66	0	4,32	4,84	0,88
22	8,56	7,16	16,41	4,84	16,63	20,08	19,26	8,91	9,49	9,66	9,61	1,94	31,27	4,02	0,71	3,07	0,78	3,60	6,53	6,83	6,62	4,32	0	6,04	4,40
23	9,38	12,47	22,45	7,56	10,64	20,51	15,83	2,88	5,45	5,31	5,36	5,85	28,75	7,28	6,24	7,38	5,93	7,54	0,53	1,34	6,99	4,84	6,04	0	5,71
24	4,41	11,29	18,86	8,96	14,82	16,07	15,09	7,90	10,90	10,86	10,88	6,01	26,88	1,96	5,10	2,82	3,65	2,57	6,19	7,01	2,23	0,88	4,40	5,71	0

### A-5) Roteiros da situação real



#### A-6) Roteiros da solução obtida



## **ANEXO B – CÓDIGO EM VBA DO MODELO DE ROTEIRIZAÇÃO**

O Anexo B apresenta o código criado em VBA (*Visual Basic for Applications*), para o algoritmo de varredura de Gillett e Miller (1974), para funcionamento na plataforma *Excel*, em conjunto com o software de otimização “*What’s Best!*”.

As rotinas principais e subrotinas são mostradas a seguir.

### B-1) Rotina de resolução do problema de caixeiro viajante (TSP)

```

Option Explicit

Dim I, J As Integer           'índices de referência
Dim lin As Integer
Dim nLinhas As Integer
Dim nColunas As Integer
Dim colAtual As Integer
Public colUi As Integer      'refere-se à coluna onde estão as variáveis Ui
Dim Xij() As Integer          'vetor que armazena as variáveis do problema
Dim rngXij As Range           'intervalo das variáveis

Public Sub TSPsolver()
'-----
'rotina principal de resolução do prob de caixeiro viajante (TSP)
'os procedimentos "wbAdjust", "WBBIN", "wbSolve", "wbBest", "wbConstraint"
'são procedimentos próprios do suplemento "What's Best!"
'-----

Worksheets("TSP").Activate
Range("B1:ET2000").ClearContents
wbAdjust "J3:DE53", "Reset"

'chama a subrotina que escreve o TSP (índices, variáveis e custos)
'-----
VarWriter

'ajusta a referência das colunas para escrever as restrições na planilha
'-----
    colAtual = 10
'chama a subrotina que escreve as restrições do TSP
'-----
    ConstraintWriter
    colAtual = colAtual + 3

'chama a subrotina que escreve as restrições de eliminação de subrotas
'de Miller-Tucker-Zemlin
'-----
    ConstrMTZ

'resolve o TSP escrito através do "What's Best!"
'-----
    wbSolve

End Sub

Private Sub VarWriter()
'-----
'escreve as variáveis do problema de caixeiro viajante (TSP) em planilha
Excel
'para as rotas determinadas na varredura
'-----

Dim indices() As Integer
Dim refCustos As Range
Dim custos As Range

```

```

lin = 4
nLinhas = (nPontos + 1) ^ 2
ReDim indices(nLinhas, 2)
Set refCustos = Worksheets("Distancias").Range("B2")

Cells(3, "B").Value = "i"
Cells(3, "C").Value = "j"
Cells(3, "E").Value = "Variáveis"
Cells(3, "G").Value = "Custos"

' escreve os índices das variáveis do TSP e os correspondentes custos
' -----
For I = 0 To nPontos
    indices(I, 1) = I
    For J = 0 To nPontos
        indices(I, 2) = J
        If I = 0 Then
            Cells(lin, "B").Value = indices(I, 1)
            If J = 0 Then
                Cells(lin, "C").Value = indices(I, 2)
            Else
                Cells(lin, "C").Value = indices(I, 2) + refInic - 1
            End If
        ElseIf J = 0 Then
            Cells(lin, "B").Value = indices(I, 1) + refInic - 1
            Cells(lin, "C").Value = indices(I, 2)
        Else
            Cells(lin, "B").Value = indices(I, 1) + refInic - 1
            Cells(lin, "C").Value = indices(I, 2) + refInic - 1
        End If
        Cells(lin, "E").Value = 0
        Cells(lin, "G").Value = refCustos.Offset(Cells(lin, "B").Value,
Cells(lin, "C").Value)
        lin = lin + 1
    Next J
Next I

'transforma o intervalo desejado em variaveis binarias
' -----
Set rngXij = Range(Cells(4, "E"), Cells(lin - 1, "E"))
WBBIN "Xij", rngXij

'renomeia o intervalo com os custos
' -----
Set custos = Worksheets("TSP").Range(Cells(4, "G"), Cells(lin - 1, "G"))
ActiveWorkbook.Names.Add Name:="Custos", RefersTo:=custos

'define a função objetivo
' -----
Application.Range("A2").Formula = "=SUMPRODUCT(Custos,WBBINXij)"
wbBest Range("A2"), "minimize"

End Sub

Private Sub ConstraintWriter()
'
```

```

' escreve as restrições do problema de caixeiro viajante
'-----

Dim restXij() As Integer
Dim restXji() As Integer
Dim indLetra As Single
Dim strColuna As String
Dim somaProd As String

nColunas = nPontos
nLinhas = (nPontos + 1) ^ 2
ReDim restXij(nLinhas, nColunas)

' escreve as restrições de arco único de entrada Xij
'-----
For J = 0 To nColunas
    lin = 3
    Cells(2, colAtual).Value = J + refInic - 1
    If J = 0 Then Cells(2, colAtual).Value = J
        For I = 0 To nLinhas
            lin = lin + 1
            If Cells(lin, "C") = Cells(2, colAtual) And Cells(lin, "B") <>
Cells(lin, "C") Then
                restXij(I, J) = 1
                Cells(lin, colAtual).Value = restXij(I, J)
            Else
                restXij(I, J) = 0
                Cells(lin, colAtual).Value = restXij(I, J)
            End If
        Next I
        If 64 + colAtual > 90 Then
            indLetra = colAtual / 26
            indLetra = Int(indLetra - 0.00001)
            strColuna = Chr(64 + indLetra) & Chr(64 + colAtual - 26 * indLetra)
        Else
            strColuna = Chr(64 + colAtual)
        End If
        somaProd = "=sumproduct(" & strColuna & "4:" & strColuna & Trim(Str(lin -
1)) & ",WBBINXij)"
        Cells(lin, colAtual).Formula = somaProd
        wbConstraint Cells(lin, colAtual), "=", "1", Cells(lin + 1, colAtual)
        colAtual = colAtual + 1
    Next J

    colAtual = colAtual + 3
    ReDim restXji(nLinhas, nColunas)

' escreve as restrições de arco único de saída Xji
'-----
For J = 0 To nColunas
    lin = 3
    Cells(2, colAtual).Value = J + refInic - 1
    If J = 0 Then Cells(2, colAtual).Value = J
        For I = 0 To nLinhas
            lin = lin + 1
            If Cells(lin, "B") = Cells(2, colAtual) And Cells(lin, "B") <>
Cells(lin, "C") Then
                restXji(I, J) = 1
                Cells(lin, colAtual).Value = restXji(I, J)
            End If
        Next I
    Next J

```

```

    Else
        restXji(I, J) = 0
        Cells(lin, colAtual).Value = restXji(I, J)
    End If
Next I
If 64 + colAtual > 90 Then
    indLetra = colAtual / 26
    indLetra = Int(indLetra - 0.00001)
    strColuna = Chr(64 + indLetra) & Chr(64 + colAtual - 26 * indLetra)
Else
    strColuna = Chr(64 + colAtual)
End If
somaProd = "=sumproduct(" & strColuna & "4:" & strColuna & Trim(Str(lin - 1)) & ",WBBINXij)"
Cells(lin, colAtual).Formula = somaProd
wbConstraint Cells(lin, colAtual), "<=", "1", Cells(lin + 1, colAtual)
colAtual = colAtual + 1
Next J

End Sub

Private Sub ConstrMTZ()
'-----
' escreve as restrições de eliminação de subtours
' da formulação de Miller-Tucker-Zemlin (MTZ)
'-----

Dim rngUi As Range           'intervalo da variável Ui
Dim strColuna As String
Dim strFormula As String
Dim indLetra As Single

colUi = colAtual + 3
lin = 4
nLinhas = (nPontos + 1) ^ 2

' escreve as variáveis Ui
'
Cells(lin, colAtual).Value = 0
Cells(lin, colAtual + 3).Value = 1
For I = 1 To nPontos
    lin = lin + 1
    Cells(lin, colAtual).Value = I + refInic - 1
    Cells(lin, colAtual + 3).Value = 0
    Cells(lin, colAtual + 5).Value = nPontos + 1
    wbConstraint Cells(lin, colAtual + 3), ">=", "2", Cells(lin, colAtual + 2)
    wbConstraint Cells(lin, colAtual + 3), "<=", Cells(lin, colAtual + 5),
    Cells(lin, colAtual + 4)
Next I

colAtual = colAtual + 3
Set rngUi = Range(Cells(5, colAtual), Cells(lin, colAtual))
wbAdjust rngUi

colAtual = colAtual + 4
lin = 3

```

```

' escreve as restrições de eliminação de subrotas
'-----
Cells(2, colAtual).Value = "Xij"
Cells(2, colAtual + 1).Value = "Ui"
Cells(2, colAtual + 2).Value = "Uj"
Cells(2, colAtual + 4).Value = "LE"
Cells(2, colAtual + 6).Value = "LD"

For I = 0 To nLinhas
    lin = lin + 1
    If Cells(lin, "B") <> Cells(lin, "C") And Cells(lin, "B") <> 0 And
Cells(lin, "C") <> 0 Then
        strFormula = "=" & Cells(lin, "E").Address
        Cells(lin, colAtual).Formula = strFormula

        For J = 0 To nPontos
            If Cells(lin, "B") = Cells(J + 4, colUi - 3) Then
                strFormula = "=" & Cells(J + 4, colUi).Address
                Cells(lin, colAtual + 1).Formula = strFormula
            End If
            If Cells(lin, "C") = Cells(J + 4, colUi - 3) Then
                strFormula = "=" & Cells(J + 4, colUi).Address
                Cells(lin, colAtual + 2).Formula = strFormula
            End If
        Next J

        strFormula = "=" & Cells(lin, colAtual + 1).Address & "-" &
Cells(lin, colAtual + 2).Address & "+1"
        Cells(lin, colAtual + 4).Formula = strFormula

        strFormula = "=" & Trim(Str(nPontos)) & "*(" & Cells(lin,
colAtual).Address & ")"
        Cells(lin, colAtual + 6).Formula = strFormula

        wbConstraint Cells(lin, colAtual + 4), "<=", Cells(lin, colAtual +
6), Cells(lin, colAtual + 5)

    End If
Next I

End Sub

```

## B-2) Rotina principal

```

Option Explicit

Public nPontos As Integer
Public refInic As Integer           'referência do ponto inicial para o veículo
v
Dim colResp As Integer
Dim colRespAnt As Integer
Dim colMelhorResp As Integer
Dim resp() As Double                'matriz que armazena a resposta de TSP
resolvido
Dim valorObj As Double              'armazena o valor atual da resposta
Dim melhorObj As Double             'armazena a melhor resposta
Dim rngMelhorObj As Range
Dim rngMelhorResp As Range
Dim tempoInic As Date               'tempo de inicio
Dim tempoTotal As Double

Sub Main()
'-----
'rotina principal do algoritmo de varredura
'esta rotina executa as rotações dos eixos para inicio da varredura
'e chama as subrotinas de varredura e mostra a resposta final na planilha
"Solução"
'-----

Dim I As Integer
Dim cont1, cont2, cont3, cont4 As Integer
Dim linAtual As Integer

tempoInic = Now
colResp = 0
linAtual = 3
Worksheets("Respostas").Activate
Range("A2:Z100").ClearContents

Worksheets("Dados").Activate

Range("A3").Select
    Range(Selection, Selection.End(xlDown)).Select
    Selection.Copy
    Range("N3").Select
    Selection.PasteSpecial      Paste:=xlPasteValues,          Operation:=xlNone,
SkipBlanks _:=False, Transpose:=False

'classifica os pontos em ordem crescente de ângulo de coord polar
'-----
Range("B3:N3").Select
    Range(Selection, Selection.End(xlDown)).Select
    Selection.Sort Key1:=Range("I3"), Order1:=xlAscending, Header:=xlNo, _
OrderCustom:=1, MatchCase:=False, Orientation:=xlTopToBottom, _
DataOption1:=xlSortNormal

Do

```

```

Select Case Cells(I + 3, "H").Value
    Case 1
        cont1 = cont1 + 1
    Case 2
        cont2 = cont2 + 1
    Case 3
        cont3 = cont3 + 1
    Case Else
        cont4 = cont4 + 1
End Select
I = I + 1
linAtual = linAtual + 1
Loop Until Cells(I + 3, "H") = ""

'início do loop que realiza as rotações do eixo de referência
'-----
Do

'executa varredura
'-----
Sweep

Worksheets("Dados").Activate

Select Case Cells(3, "H").Value
'rotação para inicio no segundo quadrante
'-----
Case 1
    Range(Cells(3, "B"), Cells(3 + cont1 - 1, "N")).Select
    Selection.Copy
    Cells(linAtual, "B").Select
    ActiveSheet.Paste
    Range(Cells(3 + cont1, "B"), Cells(linAtual + cont1 - 1, "N")).Select
    Selection.Copy
    Cells(3, "B").Select
    ActiveSheet.Paste
    Range(Cells(linAtual, "B"), Cells(linAtual + cont1, "N")).ClearContents

'rotação para inicio no terceiro quadrante
'-----
Case 2
    Range(Cells(3, "B"), Cells(3 + cont2 - 1, "N")).Select
    Selection.Copy
    Cells(linAtual, "B").Select
    ActiveSheet.Paste
    Range(Cells(3 + cont2, "B"), Cells(linAtual + cont2 - 1, "N")).Select
    Selection.Copy
    Cells(3, "B").Select
    ActiveSheet.Paste
    Range(Cells(linAtual, "B"), Cells(linAtual + cont2, "N")).ClearContents

'rotação para inicio no quarto quadrante
'-----
Case 3
    Range(Cells(3, "B"), Cells(3 + cont3 - 1, "N")).Select

```

```

        Selection.Copy
        Cells(linAtual, "B").Select
        ActiveSheet.Paste
        Range(Cells(3 + cont3, "B"), Cells(linAtual + cont3 - 1,
"N")).Select
        Selection.Copy
        Cells(3, "B").Select
        ActiveSheet.Paste
        Range(Cells(linAtual, "B"), Cells(linAtual + cont3,
"N")).ClearContents

'rotação para retornar à posição inicial
'-----
Case 4
    Range(Cells(3, "B"), Cells(3 + cont4 - 1, "N")).Select
    Selection.Copy
    Cells(linAtual, "B").Select
    ActiveSheet.Paste
    Range(Cells(3 + cont4, "B"), Cells(linAtual + cont4 - 1,
"N")).Select
    Selection.Copy
    Cells(3, "B").Select
    ActiveSheet.Paste
    Range(Cells(linAtual, "B"), Cells(linAtual + cont4,
"N")).ClearContents
End Select
Loop Until Cells(3, "H").Value = 1
'fim do loop - termina quando retorna à pos incial
'-----

'calcula tempo total de computação
'-----
tempoTotal = Val(DateDiff("s", tempoInic, Now))
Range("O5").Value = tempoTotal

OutputSol

End Sub

```

```

Sub Sweep()
'-----
'rotina que executa a etapa de varredura de todos os pontos,
'chama a rotina de resolução dos TSP formados (sub "TSPsolver")
'chama a rotina que escreve a resposta do TSP resolvido (sub "StoreAnsTSP")
'e chama a rotina que verifica se a resposta da varredura realizada é melhor
que a resp atual
'(Sub "BestAns")
'-----


Dim I As Integer
Dim nVeiculos As Integer
Dim nIteracoes As Integer
Dim outraIteracao As Boolean

nVeiculos = Cells(9, "O").Value
outraIteracao = True
nIteracoes = 0
I = 0

'escreve matriz de distâncias
'-----
DePara

'Resolução dos TSP's
'-----
Do While outraIteracao
    refInic = Worksheets("Dados").Cells(I + 3, "R").Value
    nPontos = Worksheets("Dados").Cells(I + 3, "S").Value

    TSPsolver

    StoreAnsTSP

    nIteracoes = nIteracoes + 1
    If nIteracoes >= nVeiculos Then
        outraIteracao = False
    End If
    I = I + 1

Loop

'Reposta do algoritmo
'-----
BestAns

End Sub

```

```
Private Sub StoreAnsTSP()
'-----
' armazena a resposta do TSP calculado para o veículo v da varredura atual
' escreve a resposta na planilha ("Respostas")
'-----

Dim I, J As Integer
Dim lin As Integer
Dim nLinhas As Integer

Worksheets("TSP").Activate 'Planilha TSP
nLinhas = (nPontos + 1) ^ 2
lin = 4
ReDim resp(nPontos, 1)

valorObj = Range("A2").Value
For I = 0 To nPontos
    resp(I, 0) = Cells(lin, colUi - 3).Value
    resp(I, 1) = Cells(lin, colUi).Value
    lin = lin + 1
Next I

Worksheets("Respostas").Activate 'Planilha respostas
Cells(2, colResp + 1).Value = valorObj
For I = 0 To nPontos
    Cells(I + 3, colResp + 1).Value = resp(I, 0)
    Cells(I + 3, colResp + 2).Value = resp(I, 1)
Next I
colResp = colResp + 3

End Sub
```

```

Private Sub BestAns()
'-----
'verifica se a resposta da varredura realizada é melhor do que a
'reposta atualmente armazenada
'se for melhor, armazena
'se for pior, mantém a anterior
'-----

Dim I As Integer

Worksheets("Respostas").Activate
colRespAnt = colResp - 5

If colRespAnt = 1 Then
    melhorObj = Cells(1, "A").Value
    Set rngMelhorObj = Range("A1")

    'ajuste para corrigir os pontos renumerados para a numeração original
    '-----
    Worksheets("Dados").Activate
    Range("N3").Select
    Range(Selection, Selection.End(xlDown)).Select
    Selection.Copy
    Worksheets("Respostas").Activate
    Range("J36").Select
    ActiveSheet.Paste
End If

If colRespAnt > 1 Then
    If Cells(1, colResp - 5).Value < melhorObj And Cells(1, colResp - 5).Value < Cells(1, colRespAnt - 6).Value Then
        melhorObj = Cells(1, colResp - 5)
        Set rngMelhorObj = Cells(1, colResp - 5)
    ElseIf Cells(1, colRespAnt - 6) < melhorObj Then
        melhorObj = Cells(1, colRespAnt - 6)
        Set rngMelhorObj = Cells(1, colRespAnt - 6)
    End If

    'ajuste para corrigir os pontos renumerados para a numeração original
    '-----
    Worksheets("Dados").Activate
    Range("N3").Select
    Range(Selection, Selection.End(xlDown)).Select
    Selection.Copy
    Worksheets("Respostas").Activate
    Range("J36").Select
    ActiveSheet.Paste
End If

'escreve a resposta na planilha ("Respostas")
'-----
colMelhorResp = rngMelhorObj.Column
Cells(34, "A").Value = "Melhor Resposta"
Cells(35, "A").Value = melhorObj
Set rngMelhorResp = Range(Cells(3, colMelhorResp), Cells(30, colMelhorResp + 4))
rngMelhorResp.Select
Selection.Copy

```

```
Range("A36").Select
ActiveSheet.Paste
Range("A1").Select

Worksheets("Dados").Activate
    Range("A3").Select
    Range(Selection, Selection.End(xlDown)).Select
    Selection.Copy
Worksheets("Respostas").Activate
    Range("I36").Select
    Selection.PasteSpecial      Paste:=xlPasteValues,      Operation:=xlNone,
SkipBlanks _
:=False, Transpose:=False

End Sub
```

```

Private Sub OutputSol()
'-----
' escreve a melhor resposta obtida na planilha ("Solução")
'-----

Worksheets("Solução").Activate
Range("B5:B50").ClearContents
Range("K5:K50").ClearContents

Worksheets("Respostas").Activate

'classifica os pontos da solução na ordem em que são visitados
'-----
Range("A36").Select
Range(Selection, Selection.End(xlToRight)).Select
Range(Selection, Selection.End(xlDown)).Select
Selection.Sort Key1:=Range("B36"), Order1:=xlAscending, Header:=xlGuess,
-
OrderCustom:=1, MatchCase:=False, Orientation:=xlTopToBottom, _
DataOption1:=xlSortNormal

Range("D36").Select
Range(Selection, Selection.End(xlToRight)).Select
Range(Selection, Selection.End(xlDown)).Select
Selection.Sort Key1:=Range("E36"), Order1:=xlAscending, Header:=xlGuess,
-
OrderCustom:=1, MatchCase:=False, Orientation:=xlTopToBottom, _
DataOption1:=xlSortNormal

Range("A36").Select
Range(Selection, Selection.End(xlDown)).Select
Selection.Copy
Worksheets("Solução").Activate
Range("B5").Select
ActiveSheet.Paste

'escreve as respostas, copiando-as da planilha ("Respostas")
'e colando-as na planilha ("Solução")
'-----
Worksheets("Respostas").Activate
Range("D36").Select
Range(Selection, Selection.End(xlDown)).Select
Selection.Copy
Worksheets("Solução").Activate
Range("K5").Select
ActiveSheet.Paste
Cells(2, "C").Value = Worksheets("Respostas").Cells(35, "A").Value

End Sub

```

```

Private Sub DePara()
'-----
' escreve a matriz de distâncias de acordo com o remanejamento dos pontos
' na planilha ("Distâncias")
' é executada a cada iteração da varredura
'-----

Dim I, J As Integer
Dim latA, latB As Double
Dim longA, longB As Double
Dim distIJ() As Double
Dim nLinhas As Integer

Worksheets("Dados").Activate 'Planilha dados

Do
    nLinhas = I
    I = I + 1
Loop Until Cells(I, "A").Value = ""

nPontos = nLinhas - 2

ReDim distIJ(nPontos, nPontos)

For I = 0 To nPontos
    latA = Cells(I + 2, "C")
    longA = Cells(I + 2, "D")
    For J = 0 To nPontos
        latB = Cells(J + 2, "C")
        longB = Cells(J + 2, "D")
        distIJ(I, J) = DIST(latA, latB, longA, longB)
    Next J
Next I

Worksheets("Distancias").Activate 'Planilha distancias
Range("A1:Z500").ClearContents
For I = 0 To nPontos
    Cells(I + 2, "A").Value = I
    For J = 0 To nPontos
        Cells(1, J + 2).Value = J
        Cells(I + 2, J + 2).Value = distIJ(I, J)
    Next J
Next I

End Sub

```

```
Public Function DIST(ByVal latA As Double, ByVal latB As Double, ByVal longA  
As Double, ByVal longB As Double) As Double  
'-----  
' função que calcula as distâncias entre dois pontos com a  
'fórmula de distância de grande círculo  
'-----  
  
Const FatorCircuito = 1.23  
Const PI = 3.141592654  
Dim C As Double  
Dim A As Double  
Dim DistCirc As Double  
  
C = 180 / PI  
A = Sin(latA / C) * Sin(latB / C) + Cos(latA / C) * Cos(latB / C) *  
Cos(Abs(longB - longA) / C)  
DistCirc = 6377 * Atn(Sqr(1 - A ^ 2) / A)  
DIST = FatorCircuito * DistCirc  
  
End Function
```

### B-3) Rotina de construção do gráfico de localização dos pontos

```

Option Explicit

Sub Graf()
'
'-----'
'este procedimento desenha o gráfico com a localização
'geográfica dos pontos do problema
'-----'
'

Dim I As Integer
Dim nPontos As Integer

Sheets("Localização").Select
ActiveWindow.SelectedSheets.Delete
Worksheets("Dados").Activate
nPontos = Worksheets("Dados").Cells(3, "O").Value

'-----'
'adiciona dados
'-----'

    Charts.Add
    ActiveChart.ChartType = xlXYScatter

    ActiveChart.SeriesCollection(1).XValues = "=Dados!R2C6"
    ActiveChart.SeriesCollection(1).Values = "=Dados!R2C7"
    ActiveChart.SeriesCollection(1).Name = "=Dados!R2C1"

    For I = 2 To nPontos + 1
        ActiveChart.SeriesCollection.NewSeries
        ActiveChart.SeriesCollection(I).XValues = "=Dados!R" & I + 1 & "C6"
        ActiveChart.SeriesCollection(I).Values = "=Dados!R" & I + 1 & "C7"
        ActiveChart.SeriesCollection(I).Name = "=Dados!R" & I + 1 & "C1"
    Next I

    ActiveChart.Location Where:=xlLocationAsNewSheet, Name:="Localização"
'-----'
'formatação do gráfico
'-----'

    'linhas de grade
    With ActiveChart.Axes(xlCategory)
        .HasMajorGridlines = False
        .HasMinorGridlines = False
    End With
    With ActiveChart.Axes(xlValue)
        .HasMajorGridlines = False
        .HasMinorGridlines = False
    End With

    'legenda
    ActiveChart.HasLegend = False

    'área de plotagem
    ActiveChart.PlotArea.Select
    Selection.Interior.ColorIndex = xlNone

```

```

'eixos
ActiveChart.Axes(xlCategory).Select
With Selection.TickLabels.Font
    .ColorIndex = 2
    .Background = xlTransparent
End With

ActiveChart.Axes(xlValue).Select
With Selection.TickLabels.Font
    .ColorIndex = 2
    .Background = xlTransparent
End With

'-----
'muda formatos dos pontos do gráfico
'-----
ActiveChart.SeriesCollection(1).Select
With Selection.Border
    .Weight = xlHairline
    .LineStyle = xlNone
End With
With Selection
    .MarkerBackgroundColorIndex = 3
    .MarkerForegroundColorIndex = 3
    .MarkerStyle = xlX
    .Smooth = False
    .MarkerSize = 7
    .Shadow = False
End With

'formata pontos de demanda
For I = 2 To nPontos + 1
    ActiveChart.SeriesCollection(I).Select
    With Selection.Border
        .Weight = xlHairline
        .LineStyle = xlNone
    End With
    With Selection
        .MarkerBackgroundColorIndex = 11
        .MarkerForegroundColorIndex = 11
        .MarkerStyle = xlDiamond
        .Smooth = False
        .MarkerSize = 7
        .Shadow = False
    End With
    ActiveChart.SeriesCollection(I).ApplyDataLabels           AutoText:=True,
LegendKey:= _
    False, ShowSeriesName:=True
Next I
End Sub

```