

UNIVERSIDADE ESTADUAL DE PONTA GROSSA  
SETOR DE CIÊNCIAS AGRÁRIAS E DE TECNOLOGIA  
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO APLICADA

APLICAÇÃO DE HEURÍSTICAS E META-HEURÍSTICAS NO  
DESENVOLVIMENTO DE UM SISTEMA DE APOIO A DECISÃO PARA  
RESOLUÇÃO DE PROBLEMAS DE ROTEAMENTO DE VEÍCULOS  
APLICADOS À AGRICULTURA

ROBSON FERNANDO DUDA

PONTA GROSSA

2014

ROBSON FERNANDO DUDA

APLICAÇÃO DE HEURÍSTICAS E META-HEURÍSTICAS NO  
DESENVOLVIMENTO DE UM SISTEMA DE APOIO A DECISÃO PARA  
RESOLUÇÃO DE PROBLEMAS DE ROTEAMENTO DE VEÍCULOS  
APLICADOS À AGRICULTURA

Dissertação de mestrado apresentada  
ao programa de Pós-Graduação em  
Computação Aplicada da Universidade  
Estadual de Ponta Grossa, como parte  
dos requisitos para a obtenção do título  
de mestre em Computação Aplicada na  
Área Tecnologias para Agricultura.

Orientador: Prof. Dr. Ivo Mário Mathias

PONTA GROSSA

2014

**Ficha Catalográfica**  
**Elaborada pelo Setor de Tratamento da Informação BICEN/UEPG**

D844 Duda, Robson Fernando  
Aplicação de heurísticas e meta-heurísticas no desenvolvimento de um sistema de apoio a decisão para resolução de problemas de roteamento de veículos aplicados a agricultura/ Robson Fernando Duda. Ponta Grossa, 2014.  
75f.

Dissertação (Mestrado em Computação Aplicada - Área de Concentração: Computação para Tecnologias em Agricultura), Universidade Estadual de Ponta Grossa.

Orientador: Prof. Dr. Ivo Mário Mathias.

1.Clark e Wright. 2.Sweep. 3.Tabu Search. 4.SIG. 5.PRV. I.Mathias, Ivo Mário. II. Universidade Estadual de Ponta Grossa. Mestrado em Computação Aplicada. III. T.

CDD: 005.113

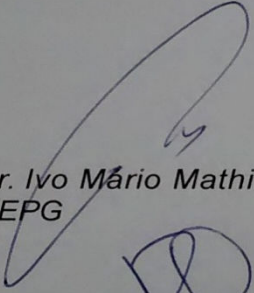
## TERMO DE APROVAÇÃO

**Robson Fernando Duda**

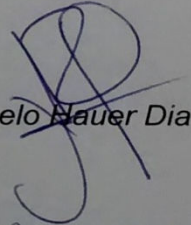
**“APLICAÇÃO DE HEURÍSTICAS E META-HEURÍSTICAS NO DESENVOLVIMENTO DE UM SISTEMA DE APOIO A DECISÃO PARA RESOLUÇÃO DE PROBLEMAS DE ROTEAMENTO DE VEÍCULOS APLICADOS À AGRICULTURA”**

Dissertação aprovada como requisito parcial para obtenção do grau de Mestre no Programa de Pós-Graduação em Computação Aplicada da Universidade Estadual de Ponta Grossa, pela seguinte banca examinadora:

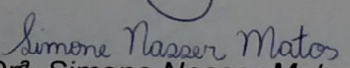
Orientadora:



*Dr. Ivo Mário Mathias*  
UEPG



*Dr. Ariangelo Bauer Dias*  
UEPG



*Simone Nasser Matos*  
Dr.<sup>a</sup> Simone Nasser Matos  
UTFPR

**Ponta Grossa, 28 de fevereiro de 2014.**

Dedico aos meus pais, Josélia e Oscar Duda, pela oportunidade de estar aqui, hoje, concluindo esse trabalho. Sem eles, nada disso seria possível.

À minha amiga, noiva e companheira Stefani por ficar ao meu lado e me incentivar, nos momentos de alegria e principalmente nos de tristeza e incerteza. Essa dissertação também é sua.

## AGRADECIMENTOS

A Deus, por me dar força e clareza de pensamentos sempre que foi necessário.

À Stefani Valéria Fischer, minha noiva, minha amiga, minha revisora, ou seja, tudo que eu precisei e sempre vou precisar para poder alcançar conquistas como essa. Sem ela, esse trabalho não seria concluído com a mesma qualidade.

Aos meus pais, eternos incentivadores, os quais sempre deram todo o apoio necessário para sempre seguir em frente. Muitas vezes acabei privado da sua companhia, mas vocês entenderam e deram apoio incondicional na minha jornada.

Ao professor Ivo Mário Mathias, meu orientador, pelas ideias, sugestões e correções que deram a esse trabalho a qualidade necessária. Mas principalmente por acreditar em mim e aceitar o desafio de me orientar e realmente fazer isso de maneira eficiente.

Ao meu colega de laboratório, amigo e orientado de iniciação científica, Luiz Antonio Zanlorensi Junior, por toda a ajuda e também pelas boas risadas, o que garantiu descontração em horas mais difíceis.

Aos meus amigos, grandes companheiros em todas as horas, que sempre me deram força e incentivo para seguir em frente. Em especial a Daniel e Viviane Zadra, amigos de sempre, a Patrícia Blum, revisora e amiga e a todos que me ajudaram de uma forma ou outra.

A Cooperativa Castrolanda, pela possibilidade de observar um ambiente real de distribuição de produtos que deu origem aos resultados desse trabalho. Também a Renato Leffers, nosso guia na empresa, que não poupou esforços em nos ajudar e permitir que esse trabalho fosse concluído.

Agradeço a CAPES pelo apoio financeiro.

A mente que se abre a uma  
nova ideia jamais voltará a seu  
tamanho original.

(Albert Einstein)

## RESUMO

Este trabalho apresenta uma solução para o problema de roteamento de veículos com frotas homogêneas. Para tanto, foram desenvolvidos algoritmos baseados em heurísticas e meta-heurísticas aplicadas ao desenvolvimento de um sistema de apoio a decisão, com interface georreferenciada. Os algoritmos tiveram como base métodos heurísticos construtivos e em duas fases, além de uma meta-heurística. A camada de interface utilizada como componente de visualização é baseada em dados cartográficos que indicam a localização dos pontos a serem atendidos e as vias que os interligam, formando a malha viária que é representada utilizando a API do *Google Maps*<sup>®</sup>. Os algoritmos foram validados utilizando instâncias da literatura, apresentando resultados satisfatórios em relação a otimização baseada nos métodos utilizados, mostrando ser possível a utilização do sistema desenvolvido para a distribuição de produtos agrícolas.

**Palavras-chave:** *Clark e Wright, Sweep, Tabu Search, SIG, PRV.*



## ABSTRACT

This paper presents a solution to the routing problem of vehicles with homogeneous fleet. To do so, heuristic and metaheuristic based algorithms applied towards the development of a decision support system, with georeferenced interface were developed. The algorithms had as base heuristic methods built in two phases, besides a metaheuristic. The interface layer used as visualization component is based in cartographic data that indicates the location of the points to be assisted and the paths that connects them, forming a road system represented using the *Google Maps*<sup>®</sup> API. The algorithms were validated using instances from the literature, presenting satisfactory results regarding optimization based in the methods that were used, showing that it is possible the usage of the developed system in the distribution of agricultural products.

**Keywords:** *Clark e Wright, Sweep, Tabu Search, SIG, PRV.*

## LISTA DE ILUSTRAÇÕES

|                                                                                                 |    |
|-------------------------------------------------------------------------------------------------|----|
| Figura 1 - Cenário de distribuição para um PRV .....                                            | 26 |
| Figura 2 - Resultado da geração de rotas para um PRV .....                                      | 27 |
| Figura 3 - Classificação dos métodos de solução para PRV .....                                  | 28 |
| Figura 4 - Espaço de soluções para um problema de otimização. ....                              | 29 |
| Figura 5 - Movimentos de melhoria intrarrotas .....                                             | 30 |
| Figura 6 - Movimento de melhoria inter-rotas .....                                              | 31 |
| Figura 7 - Primeira etapa do processo de economias de <i>Clark e Wright</i> .....               | 32 |
| Figura 8 - Segunda etapa do processo de economias de <i>Clark e Wright</i> (1962).....          | 33 |
| Figura 9 - Semirreta gerada com base no sistema de coordenadas polares .....                    | 34 |
| Figura 10 - Processo de <i>clusterização</i> .....                                              | 36 |
| Figura 11 - Fluxograma do algoritmo <i>Sweep</i> . ....                                         | 36 |
| Figura 12 - Algoritmo de definição da meta-heurística <i>Tabu Search</i> . ....                 | 39 |
| Figura 13 - Padrão de arquivos para PRV .....                                                   | 44 |
| Figura 14 - Visão geral do sistema de validação de algoritmos heurísticos.....                  | 47 |
| Figura 15 - Opções de roteamento do sistema .....                                               | 48 |
| Figura 16 - Tela de exibição de resultados dos algoritmos .....                                 | 49 |
| Figura 17 - Visualização gráfica dos resultados de otimização .....                             | 49 |
| Figura 18 - Gráfico dos resultados para as instâncias de Fisher.....                            | 50 |
| Figura 19 - Gráfico de tempo para as instâncias de Fisher .....                                 | 51 |
| Figura 20 - Gráfico dos resultados para as instâncias de Christofides e Eilon (1969). 53        |    |
| Figura 21 - Gráfico de tempo para as instâncias de Christofides e Eilon (1969). ....            | 54 |
| Figura 22 - Gráfico de resultado para as instâncias de Augerat et al. (1995).....               | 55 |
| Figura 23 - Gráfico de tempo para as instâncias de Augerat et al. (1995).....                   | 56 |
| Figura 24 - Arquitetura do sistema .....                                                        | 58 |
| Figura 25 - Arquitetura do sistema .....                                                        | 58 |
| Figura 26 - Diagrama geral de casos de uso .....                                                | 60 |
| Figura 27 - Diagrama de casos de uso .....                                                      | 62 |
| Figura 28 - Tela de manutenção de variáveis de adjacência.....                                  | 64 |
| Figura 29 - Visualização na camada SIG do sistema sem roteamento das entregas. . 65             |    |
| Figura 30 - Visualização da camada SIG usando o modo de mapas. ....                             | 65 |
| Figura 31 - Representação de ligação entre pontos dentro de uma rota. ....                      | 66 |
| Figura 32 - Resultado da operação de atualização de coordenadas utilizando a interface SIG..... | 67 |

## LISTA DE QUADROS

|                                                                            |    |
|----------------------------------------------------------------------------|----|
| Quadro 1 - Detalhamento das instâncias de Augerat et al. (1995) .....      | 42 |
| Quadro 2 - Detalhamento das instâncias de Christofides e Eilon (1969)..... | 43 |
| Quadro 3 - Detalhamento das Instâncias de Fisher (1994) .....              | 43 |
| Quadro 4 - Relação de telas do sistema .....                               | 63 |

## LISTA DE TABELAS

|                                                                                                                          |    |
|--------------------------------------------------------------------------------------------------------------------------|----|
| Tabela 1 - Resultado dos testes computacionais para as instâncias de Fisher .....                                        | 50 |
| Tabela 2 - Tempo computacional resultante da execução dos algoritmos em milissegundos .....                              | 51 |
| Tabela 3 - Resultado dos testes computacionais para as instâncias de Christofides e Eilon (1969). .....                  | 52 |
| Tabela 4 - Tempo de processamento para os testes computacionais para as instâncias de Christofides e Eilon (1969). ..... | 53 |
| Tabela 5 - Resultado dos testes computacionais para as instâncias de Augerat et al. (1995). .....                        | 55 |
| Tabela 6 - Tempo de processamento para os testes computacionais para as instâncias de Augerat et al. (1995). .....       | 56 |

## LISTA DE SIGLAS

|              |                                                             |
|--------------|-------------------------------------------------------------|
| <b>CSS</b>   | <i>Cascading Style Sheets</i>                               |
| <b>GPS</b>   | <i>Global Positioning System</i>                            |
| <b>HTML</b>  | <i>Hypertext Markup Language</i>                            |
| <b>JEE</b>   | <i>Java Enterprise Edition</i>                              |
| <b>JSF</b>   | <i>Java Server Faces</i>                                    |
| <b>MER</b>   | Modelo Entidade Relacionamento                              |
| <b>PCV</b>   | Problema do Caixeiro Viajante                               |
| <b>PRV</b>   | Problema de Roteamento de Veículos                          |
| <b>PRVC</b>  | Problema de Roteamento de Veículos Capacitados              |
| <b>PRVEF</b> | Problema de Roteamento de Veículos com Entregas Fracionadas |
| <b>PRVFH</b> | Problema de Roteamento de Veículos com Frota Heterogênea    |
| <b>PRVJT</b> | Problema de Roteamento de Veículos com Janelas de Tempo     |
| <b>RFCS</b>  | <i>Route First Cluster Second</i>                           |
| <b>SIG</b>   | Sistema de Informação Geográfica                            |
| <b>UML</b>   | <i>Unified Modeling Language</i>                            |
| <b>VRP</b>   | <i>Vehicle Routing Problem</i>                              |

## SUMÁRIO

|                                                                          |    |
|--------------------------------------------------------------------------|----|
| 1 INTRODUÇÃO.....                                                        | 14 |
| 1.2 OBJETIVOS.....                                                       | 16 |
| 1.1 Objetivo Geral.....                                                  | 16 |
| 1.2 Objetivos Específicos.....                                           | 16 |
| 1.3 ESTRUTURA DO TRABALHO.....                                           | 16 |
| 2 PROBLEMA DE ROTEAMENTO DE VEÍCULOS.....                                | 18 |
| 2.1 DEFINIÇÃO.....                                                       | 18 |
| 2.1.1 Conceitos Básicos Relacionados ao PRV.....                         | 18 |
| 2.1.2 Formulação do problema.....                                        | 20 |
| 2.1.3 Taxonomia e Classificação.....                                     | 22 |
| 2.1.4 Problema de Roteamento de Veículos e Suas Variações.....           | 23 |
| 3 ESTRATÉGIAS DE RESOLUÇÃO PARA PROBLEMAS DE ROTEAMENTO DE VEÍCULOS..... | 26 |
| 3.1 HEURÍSTICAS.....                                                     | 29 |
| 3.1.1 Heurística de Clark e Wright.....                                  | 31 |
| 3.1.2 Heurística de Gillet e Miller (sweep).....                         | 33 |
| 3.2 META-HEURÍSTICAS.....                                                | 37 |
| 3.2.1 <i>Tabu Search</i> .....                                           | 37 |
| 4 MATERIAIS E MÉTODOS.....                                               | 40 |
| 4.1 ALGORITMOS DESENVOLVIDOS.....                                        | 40 |
| 4.2 BASES DE DADOS.....                                                  | 41 |
| 4.3 TESTES COMPUTACIONAIS.....                                           | 43 |
| 4.3.1 Cálculo Para Distâncias Euclidianas.....                           | 45 |
| 4.3.2 Cálculo Para Distâncias Baseadas em Coordenadas Geográficas.....   | 45 |
| 5 RESULTADOS.....                                                        | 47 |
| 5.1 VALIDAÇÃO DOS ALGORITMOS DESENVOLVIDOS.....                          | 47 |
| 5.1.1 Resultados Computacionais.....                                     | 50 |

|                                                                                |    |
|--------------------------------------------------------------------------------|----|
| 5.2 SISTEMA DE APOIO A DECISÃO PARA RESOLUÇÃO DE PRV COM<br>INTERFACE SIG..... | 57 |
| 5.2.1 Arquitetura do Sistema.....                                              | 57 |
| 5.2.2 Modelagem do Sistema.....                                                | 59 |
| 5.2.3 Visão Geral do Sistema.....                                              | 62 |
| 5.2.4 Camada de Visualização SIG .....                                         | 64 |
| 6 DISCUSSÃO.....                                                               | 68 |
| 7 CONCLUSÕES.....                                                              | 70 |
| REFERÊNCIAS .....                                                              | 71 |

## 1 INTRODUÇÃO

A otimização do transporte de produtos implica em vantagens econômicas, ambientais e competitivas, que tem importante relevância no contexto econômico das empresas, pois os custos relacionados ao transporte são incorporados no valor final dos produtos. Como destaca Lima (2005) um sistema de transporte e distribuição eficiente e com custos otimizados contribui para reduzir os preços das mercadorias, com conseqüente melhoria da competitividade.

Nesse contexto, surgiu o Problema de Roteamento de Veículos (PRV), proposto por Dantzig e Ramnser (1959), que teve origem a partir da necessidade de distribuir produtos de maneira mais eficiente para todos os consumidores que demandavam seu consumo. Segundo os autores, esse problema é uma generalização do tradicional Problema do Caixeiro Viajante, onde é necessário determinar o menor caminho passando por um número determinado de pontos, atendendo a sua demanda e retornando ao ponto de origem de distribuição. O problema de roteamento de veículos é utilizado para o gerenciamento de veículos e suas atividades de entrega, provendo soluções que otimizam os custos de entrega.

A preocupação em realizar um processo de entrega com economia de recursos pode ser notada diante do crescente número de trabalhos desenvolvidos com essa temática e suas diferentes áreas de aplicação. Estudos como de Guerreiro (2009) e Viana (2007) preocupam-se em determinar algoritmos que possam diminuir os custos de distribuição de produtos de maneira satisfatória. Porém, como destaca Lima (2005), além da determinação de rotas otimizadas, outra preocupação é a viabilidade e aplicação prática das rotas geradas dentro de um ambiente real de distribuição de produtos. Essa característica pode ser vista em Campelo Junior (2010) que estudou a viabilidade na distribuição de cartas e encomendas da Empresa de Correios e Telégrafos, em Nascimento (2011) que aplicou na distribuição de refeições e em Tiburcio (2012) que aplicou no transporte de funcionários de uma empresa.

Diante disso, esse trabalho propõe o desenvolvimento de um sistema de apoio a decisão baseado em heurísticas e meta-heurísticas que possibilite



determinar rotas otimizadas, diminuindo fatores como tempo de entrega e custos com a distribuição dos produtos e que possam ser aplicadas em ambientes reais de distribuição de produtos agrícolas. Com isso, será possível tornar o processo mais eficiente, fornecendo opções de roteamento baseadas na otimização computacional do cenário de distribuição.

Uma característica relevante no sistema proposto é a utilização de interfaces gráficas que facilitem a visualização das soluções propostas. Desta forma, desenvolveu-se uma interface de visualização de mapas a partir das coordenadas geográficas dos pontos de origem e entrega de produtos. Esse tipo de recurso já foi utilizado para o processo de organização e visualização de dados relacionados a problemas de roteamento de veículos, gerando resultados satisfatórios (CASAL, 2012; SANTOS, 2012). Para isso foi escolhida a API do *Google Maps*<sup>®</sup> (GOOGLE MAPS, 2014), por fornecer informações geográficas fundamentais como a cartografia do terreno, estrutura das estradas, além de dados adicionais como restrição de tráfego e sentido das ruas.

## 1.2 OBJETIVOS

### 1.1 Objetivo Geral

A partir do estudo de técnicas de otimização combinatória, propor um sistema de apoio a decisão computacional para aplicação na otimização de rotas de distribuição de produtos agrícolas, que seja capaz de produzir rotas satisfatórias e com isso minimizar os custos relacionados às atividades de entrega desses produtos.

### 1.2 Objetivos Específicos

- Avaliar o desempenho das heurísticas e meta-heurísticas desenvolvidas para a solução de problemas de roteamento de veículos;
- Modelar e desenvolver um sistema de apoio a decisão capaz de simular rotas otimizadas que possam ser utilizadas em ambientes reais de distribuição de produtos agrícolas, como fertilizantes, defensivos e produtos acabados;
- Desenvolver uma camada de interface baseada em sistemas SIG para permitir mapeamento e definição de rotas em problemas reais que envolvam coordenadas geográficas.

## 1.3 ESTRUTURA DO TRABALHO

Esse trabalho foi dividido em 6 capítulos onde são explicadas as etapas necessárias para alcançar seus objetivos. São apresentadas as teorias necessárias para o seu entendimento, os métodos desenvolvidos e os resultados alcançados.

No primeiro capítulo, constituído por essa sessão, são inseridos os conceitos iniciais do trabalho, motivação para sua elaboração, justificativa e objetivos.

O segundo capítulo é destinado a denificação do problema de roteamento de veículos, conceitos básicos relacionados a esse problema, definição de taxonomias e classificação de acordo com as variáveis de restrição.

Na sequência, no terceiro capítulo, são apresentados os métodos de resolução utilizados em problemas de roteamento de veículos, contituídos por

heurísticas e meta-heurísticas, utilizadas como referência do desenvolvimento dos algoritmos nesse trabalho.

No quarto capítulo são definidos os materiais e métodos utilizados de modo a alcançar os objetivos. Além disso, são mostrados os testes de validação, dados utilizados e ferramentas desenvolvidas.

No quinto capítulo são apresentados os resultados e a discussão sobre o trabalho e por fim, no sexto capítulo, estão descritas as conclusões e indicações de trabalhos futuros.

## **2 PROBLEMA DE ROTEAMENTO DE VEÍCULOS**

### **2.1 DEFINIÇÃO**

O Problema de Roteamento de Veículos (PRV) ou *Vehicle Routing Problem* (VRP) foi definido por Dantzig e Ramnser (1959) quando foi relatada a utilização de algoritmos de programação matemática para um problema real de distribuição de gasolina e a partir de então passou a ser estudado e ampliado. O mesmo baseia-se em problemas reais de distribuição e a cada nova variável ou restrição, o problema passa a ganhar uma maior complexidade e a demandar novas estratégias de resolução.

Segundo Viana (2007), a definição simplificada para o problema de roteamento de veículos seria que, dada uma frota com  $k$  veículos, um conjunto de pontos  $V$  a serem visitados por esta frota e um depósito central no qual estão estacionados os veículos antes do início da jornada de trabalho. Pode-se afirmar que o PRV consiste em definir  $k$  rotas de veículos, onde uma rota é definida como sendo um ciclo que inicia no depósito central, percorre um subconjunto dos pontos  $V$  e termina, evidentemente, no depósito central.

Pode-se considerar, ainda, que um sistema de roteamento pode ser caracterizado como um conjunto organizado de meios que objetiva o atendimento de demandas localizadas nos arcos ou nos vértices de alguma rede de transporte (GOLDBARG, 2000).

As rotas originadas em um problema de roteamento de veículos são geradas de acordo com o critério de otimização escolhido e variam em relação a diferentes critérios utilizados. Esses critérios são os objetivos de minimização e geralmente são a distância total percorrida ou o tempo total gasto para a distribuição, visto que a menor distância nem sempre implica em menor tempo. Ainda, em alguns casos, pode-se combinar diferentes critérios em uma função de minimização que leve em consideração outros fatores.

#### **2.1.1 Conceitos Básicos Relacionados ao PRV**

Os principais componentes do problema são: clientes, veículos, depósitos, rede de transporte, restrições e variáveis de otimização. Segundo Casal (2012) é necessário ter conhecimento desses componentes, pois eles são a base a

partir da qual são formuladas restrições e casos particulares do problema de roteamento de veículos.

- **Clientes:** Os clientes ou consumidores são os locais a serem atendidos durante o processo de entrega. Nesses pontos são definidas as demandas, ou seja, a quantidade de produto que cada um deve receber. Eles representam os vértices de um grafo e servem, por exemplo, para restringir a carga dos veículos utilizados na distribuição, dados que os mesmos podem ter capacidade de carga máxima (CASAL, 2012; GUERREIRO, 2009).
- **Veículos:** São os meios utilizados para a realização das entregas em um PRV. Ao analisar o problema, deve-se levar em consideração basicamente a quantidade de veículos, a sua capacidade, os custos para a realização da entrega, o tempo gasto durante a percurso e o consumo de combustível. Em situações que exigem maior detalhamento, podem ser coletados dados sobre tipo de produtos que podem ser transportados, compatibilidade com tipos de vias ou se são veículos de carga e descarga (CASAL, 2012; GUERREIRO, 2009).
- **Depósitos:** Representam os pontos de origem para o atendimento da demanda dos clientes. Em relação a eles, a questão mais relevante é se as entregas partem de um único depósito ou partem de dois ou mais, pois essa característica muda completamente a estratégia de resolução do problema (CASAL, 2012; GUERREIRO, 2009).
- **Rede de Transporte:** É representada por um grafo o qual pode ser representado por  $G=(V,E)$ , onde  $V$  representa os vértices do grafo e  $E$  as arestas que interligam os vértices. Em um ambiente real de distribuição, os vértices podem representar os clientes, ou seja, os pontos a serem atendidos, os depósitos de onde originam-se as entregas e as interseções rodoviárias. Já as arestas representam a ligação entre esses pontos e são responsáveis por armazenar os dados de distâncias entre os pontos, quando a otimização é realizada levando

em consideração unicamente a distância, ou também podem armazenar dados sobre custo ou tempo de viagem. Os grafos que representam a rede de transporte podem ser direcionados ou não, dependendo das características do problema estudado. Geralmente usa-se grafos não direcionados em situações onde as escalas de distância são maiores, como cálculo entre países ou regiões e grafos direcionados em escalas menores, como dentro de cidades. Nos casos onde os custos estão associados com os vértices, a distância entre os pontos é determinada pela distância euclidiana e a matriz de distância é simétrica (CASAL, 2012; GUERREIRO, 2009).

- **Restrições:** Direcionam a escolha da técnica de resolução dos PRV, podendo ser locais, quando se analisa uma rota de maneira particular, ou globais, quando é levado em consideração o conjunto de rotas. Entre as restrições locais, as mais comuns são capacidade do veículo e distância máxima da rota. Já as globais são o número de veículos, tempo utilizado para atendimento e total do percurso percorrido em um período de tempo (CASAL, 2012; GUERREIRO, 2009).
- **Variáveis de Otimização:** Definem quais os critérios adotados para o processo de otimização. Geralmente representam os vértices dentro da rede de transporte e os valores são atribuídos de acordo com a variável escolhida. As variáveis mais utilizadas são a distância e o tempo, mas pode também existir outras variáveis derivadas de funções para ponderar o custo de deslocamento entre os vértices (CASAL, 2012; GUERREIRO, 2009).

Com o entendimento desses conceitos é possível a formulação matemática para o PRV clássico de acordo com as definições apresentadas aqui.

### 2.1.2 Formulação do problema

Por se tratar de um problema matemático baseado em restrições, o PRV requer um modelo de definição que contemple as suas restrições e objetivos. Segundo Goldberg (2000), formular o problema de roteamento de veículos não é uma tarefa trivial. Uma das formulações mais utilizadas como base de

solução é a de Fisher e Jaikumar (1981). A modelagem matemática, baseada em Goldberg (2000 apud Fisher e Jaikumar, 1981) é apresentada a seguir, onde é definida a equação de minimização e as suas restrições.

$$\text{Minimizar } z = \sum_{i,j} (c_{ij} \sum_k x_{ijk})$$

Sujeito a:

$$\sum_k y_{ik} = 1 \quad i = 2, \dots, n \quad (1)$$

$$\sum_k y_{1k} = m \quad i = 1 \quad (2)$$

$$\sum_i q_i y_{ik} \leq Q_k \quad k = 1, \dots, m \quad (3)$$

$$\sum_j x_{ijk} = \sum_j x_{jik} = y_{ik} \quad i = 1, \dots, n \quad k = 1, \dots, m \quad (4)$$

$$\sum_{i,j \in S} x_{ijk} \leq |S| - 1 \quad \forall S \subseteq \{2, \dots, n\}, k = 1, \dots, m \quad (5)$$

$$y_{ik} \in \{0,1\} \quad i = 1, \dots, n \quad k = 1, \dots, m \quad (6)$$

$$x_{ijk} \in \{0,1\} \quad i,j = 1, \dots, n \quad k = 1, \dots, m \quad (7)$$

Onde:

$x_{ijk}$  = variável binária que assume valor 1 quando o veículo  $k$  visita o cliente  $j$  imediatamente após o cliente  $i$ , o em caso contrário;

$y_{ik}$  = variável binária que assume valor 1 se o cliente  $i$  é visitado pelo veículo  $k$ , 0 em caso contrário;

$q_i$  é a demanda do cliente  $i$ ;

$Q_k$  é a capacidade do veículo  $k$ ;

$Q_i$  é o custo de percorrer o trecho que vai do cliente  $i$  ao  $j$ .

As restrições definidas na equação (1) asseguram que um veículo não visite mais de uma vez um cliente. As restrições (2) garantem que o depósito receba uma visita de todos os veículos. As restrições (3) obrigam que as capacidades dos veículos não sejam ultrapassadas. As restrições (4) garantem

que os veículos não param suas rotas em um cliente. As restrições (5) constituem as tradicionais restrições de eliminação de sub-rotas.

Essa é a modelagem tradicional do PRV, levando em consideração as restrições de um problema com uma única origem, frota heterogênea e demanda estática. Para problemas com mais de uma origem, frota heterogênea ou demanda dinâmica, se faz necessário uma modelagem que atenda a essas variações.

### **2.1.3 Taxonomia e Classificação**

Existem diferentes fatores que determinam natureza diferente ao PRV. Em seu trabalho, Bodin e Golden (1981) apresentam os principais critérios de classificação para um PRV.

- A. Tempo para servir determinado nó
  - a. Tempo especificado e prefixado
  - b. Janela de tempo
- B. Número de depósitos
  - a. Um depósito
  - b. Múltiplos depósitos
- C. Tipo da frota disponível
  - a. Homogênea
  - b. Heterogênea
- D. Tamanho da frota
  - a. Um veículo
  - b. Mais de um veículo
- E. Natureza da demanda
  - a. Determinística
  - b. Estocástica
- F. Localização da demanda
  - a. Nos vértices
  - b. Nos arcos



- G. Tipo de Grafo
  - a. Direcionado
  - b. Não direcionado
  - c. Misto
- H. Restrição da capacidade do veículo
  - a. Todos sujeitos a mesma restrição
  - b. Restrições diferentes para cada um
- I. Tempo máximo de roteamento
  - a. O mesmo para todos
  - b. Tempos diversos
- J. Custos
  - a. Fixos
  - b. Variáveis
- K. Operações
  - a. Entrega
  - b. Coleta
  - c. Misto
- L. Objetivos
  - a. Minimizar custos
  - b. Minimizar veículos
  - c. Misto

Esses critérios, além de permitirem maior detalhamento de um problema de roteamento de veículos e guiar o processo de resolução, também tornam possível dividir o mesmo em classes, mostradas a seguir.

#### **2.1.4 Problema de Roteamento de Veículos e Suas Variações**

Conforme Cunha (2003), o primeiro problema de roteirização estudado foi o Problema do Caixeiro Viajante (PCV). Este, consiste em encontrar o roteiro ou a sequência de cidades a serem visitadas por um caixeiro viajante, que minimize a distância total percorrida, assegurando que cada cidade seja visitada exatamente uma vez.

Porém, diferentemente do PCV, os problemas de roteamento de veículos possuem requisitos e restrições que são impostas sobre a construção

das rotas. Por exemplo, o serviço pode envolver tanto entregas e coletas, a capacidade de carga ao longo de cada rota não deve exceder a capacidade do caminhão, os pontos atendidos possuem limite de tempo para realizar a entrega, entre outros fatores. São essas variações que originam diferentes classes de PRV, sendo que os principais são listados a seguir.

#### **2.1.4.1 Problema de Roteamento de Veículos Capacitados**

O Problema de Roteamento de Veículos Capacitado (PRVC) é uma generalização do Problema de Roteamento de Veículos (PRV) descrito por Dantzig e Ramnser (1959). No PRVC os clientes representam as entregas e suas demandas são determinísticas, ou seja, conhecidas com antecedência. As entregas partem de um único depósito central e os veículos são idênticos, tendo apenas limitações de capacidade de carga. O objetivo é o de minimizar o custo total necessário para servir todos os clientes. Geralmente, o custo de viagem entre cada par de clientes é o mesmo em ambas as direções, ou seja, a matriz de custo resultante é simétrica, enquanto que em algumas aplicações, como a distribuição em áreas urbanas com as direções de sentido único impostas nas estradas, a matriz de custos é assimétrico (TOTH; VIGO, 2002).

#### **2.1.4.2 Problema de Roteamento de Veículos com Frota Heterogênea**

Nessa versão do PRV existe uma variação em relação ao PRVC relacionada à frota de veículos. No caso anterior, a frota era homogênea, ou seja, todos os veículos são iguais. Nesse caso, a frota de veículos é diversa e podem ter diferentes capacidades. Do ponto de vista de problemas reais de roteamento de veículos, isso é mais viável, pois as frotas são assim na maioria dos casos reais (GUERREIRO, 2009; GENDREAU, 1999).

#### **2.1.4.3 Problema de Roteamento de Veículos com Entregas Fracionadas**

Definida pela primeira vez em Dror e Trudeau (1989) o Problema de Roteamento de Veículos com Entregas Fracionadas (PRVEF) passou a ser objeto de estudo em outros trabalhos, em razão da sua aplicação prática (DROR; LAPORTE; TRUDEAU, 1994; ARCHETTI; SPERANZA, 2008). Na definição do PRV, uma frota de veículos capacitados está disponível para servir um conjunto de clientes com demanda conhecida. Cada cliente é obrigado a

ser visitado por exatamente um veículo e o objetivo é minimizar a distância total percorrida. No PRVEF, a restrição de que cada cliente deve ser visitado exatamente uma vez é removida, ou seja, entregas fracionadas são permitidas, sendo que cada cliente pode ser visitado mais de uma vez e por veículos diferentes.

#### **2.1.4.4 Problema de Roteamento de Veículos com Janelas de Tempo**

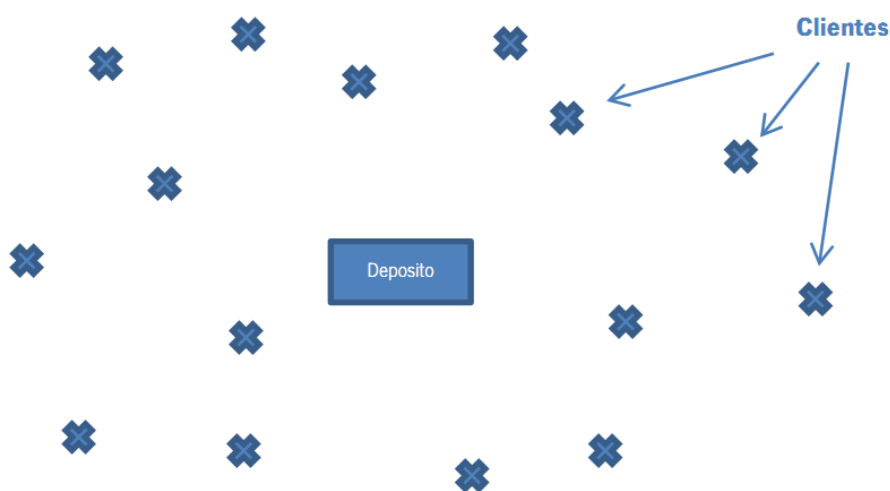
Como as demais variações do PRV, o Problema de Roteamento de Veículos com Janelas de Tempo (PRVJT) tem origem a partir da adição de uma nova restrição em relação ao PRV. Nesse caso a restrição é o tempo de operação disponível em cada ponto para recebimento, ou seja, o intervalo de tempo em que o ponto a ser atendido estará disponível para receber a entrega.

O objetivo deste problema é o de minimizar o número de veículos que constituem a frota, o tempo de viagem e o tempo de espera necessária para atender os clientes dentro da janela de tempo disponível para atendimento (CORDEAU, 2011).

### 3 ESTRATÉGIAS DE RESOLUÇÃO PARA PROBLEMAS DE ROTEAMENTO DE VEÍCULOS

No caso estudado nesse trabalho, uma frota de veículos com capacidade de  $n$  unidades de peso deverá atender a demanda existente em um conjunto de pontos a serem atendidos, partindo do depósito central e retornando ao mesmo após a execução de cada rota. Os pontos de atendimento são distribuídos ao longo de coordenadas que definem a sua localização. Esse processo está ilustrado na Figura 1.

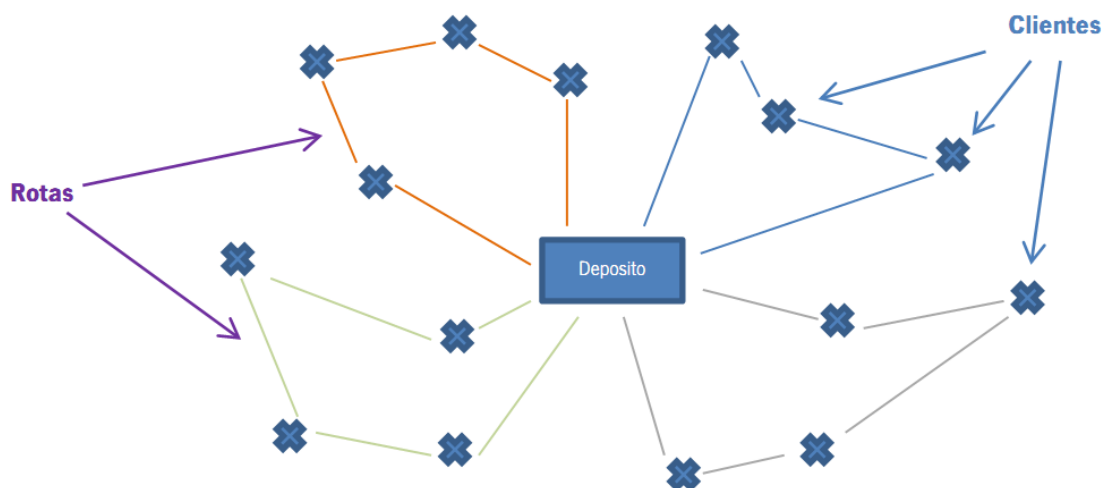
Figura 1 - Cenário de distribuição para um PRV



Fonte: Adaptado de Casal (2012)

O cenário definido acima é então submetido aos algoritmos baseados em métodos de otimização, tentando encontrar a melhor solução possível para realizar todas as entregas de forma minimizada, traçando rotas que são as trajetórias a serem seguidas, conforme Figura 2.

Figura 2 - Resultado da geração de rotas para um PRV



Fonte: Adaptado de Casal (2012)

Para alcançar o objetivo descrito acima é necessário definir qual o método de resolução a ser utilizado no problema de roteamento de veículos. Sendo assim, é necessário conhecer quais as abordagens e possíveis métodos de solução. Em seu trabalho, Belfiore (2006) faz um levantamento da complexidade computacional e das estratégias de solução para problemas de roteamento de veículos.

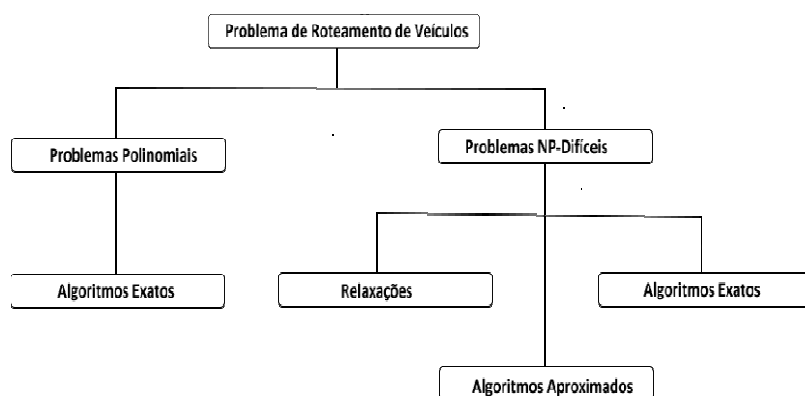
Primeiramente é necessário definir a complexidade computacional do problema tratado. Segundo Belfiore (2006) a complexidade pode ser definida como o comportamento de algoritmos para sua resolução baseia-se no número de operações necessárias para resolver um dado problema. Dessa forma, quanto mais operações o algoritmo possuir, mais lento será o processo de resolução. Nesse caso, a complexidade pode ser afetada pelo número de variáveis de restrição existentes no problema a ser resolvido, pois quanto maior o número de variáveis maior a quantidade de combinações possíveis, visto que a natureza dos problemas de roteamento de veículos é combinatória.

Diante da complexidade existente na especificação de problemas de roteamento de veículos e suas variações, os esforços para resolvê-los deram origem a diferentes estratégias de resolução. Essas estratégias podem ser classificadas em dois tipos: os métodos exatos e métodos heurísticos. Os métodos exatos exploram todo o universo de soluções, esgotando todas as possibilidades combinatórias possíveis, tendo maior aproximação de resultados ótimos (TOH e VIGO, 2002). Já os métodos heurísticos, exploram diferentes

espaços de solução baseados em mínimos locais e fornecem soluções otimizadas, mas não ótimas. A vantagem dos métodos heurísticos em relação aos métodos exatos é que seu tempo de resolução é mais baixo e as soluções não perdem em qualidade para os métodos exatos, o que faz com que a sua utilização seja mais ampla, principalmente em problemas reais de roteamento de veículos.

Mas além dessas duas abordagens supracitadas, ainda existem outros métodos matemáticos para a resolução de PRV, como a utilização de relaxações matemáticas na busca pela melhor solução. Na Figura 3 é mostrado um fluxograma com as divisões de estratégias de solução para PRV, segundo Goldberg (2000).

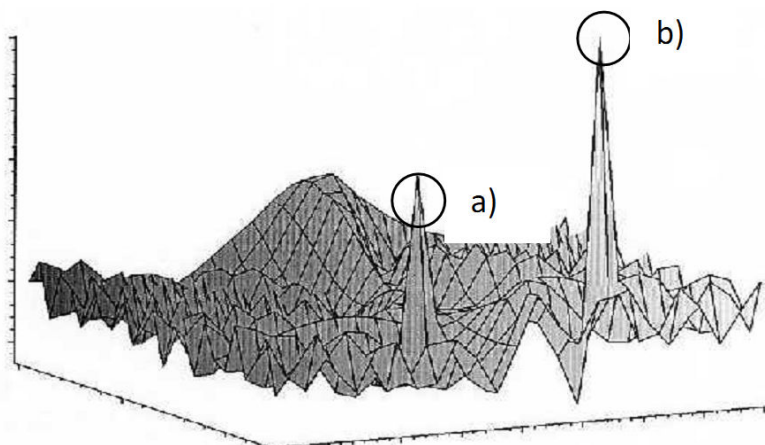
Figura 3 - Classificação dos métodos de solução para PRV



Fonte: Adaptado de Goldberg (2000)

De acordo com as características apresentadas nesse trabalho e os resultados apresentados na literatura foram escolhidos algoritmos aproximados para a resolução do PRV em questão, pois apesar de não apresentarem uma solução ótima, exploram os espaços de solução de maneira satisfatória e com um tempo computacional aceitável. O espaço de solução tratado aqui é o conjunto de soluções possíveis para um problema de roteamento de veículos, conforme Figura 4.

Figura 4 - Espaço de soluções para um problema de otimização.



Fonte: Adaptado de Guerreiro (2009)

Segundo Guerreiro (2009), levando em consideração o espaço de soluções, os métodos exatos percorrem todo o espaço de pesquisa, garantindo um ótimo global. Uma heurística segue uma regra de otimização local o que acaba deixando-a presa em ótimos locais. Já as meta-heurísticas conseguem fugir de ótimos locais, explorando melhor o espaço de pesquisa, sem, no entanto explorar todo o conjunto de soluções.

### 3.1 HEURÍSTICAS

Entre as técnicas utilizadas para a resolução do PRV destacam-se as heurísticas. Uma característica presente na utilização de métodos heurísticos é que, como destaca Belfiore (2006), elas podem ser adaptadas a problemas de roteirização diversos, como problemas de múltiplos depósitos, diferentes categorias de produtos, além da adição de novos critérios e restrições.

As técnicas heurísticas podem ser classificadas de acordo com a sua característica de funcionamento. Segundo Laporte e Semet (2001) os métodos heurísticos podem ser divididos em: heurísticas construtivas, heurística em duas fases e heurísticas de melhoramento.

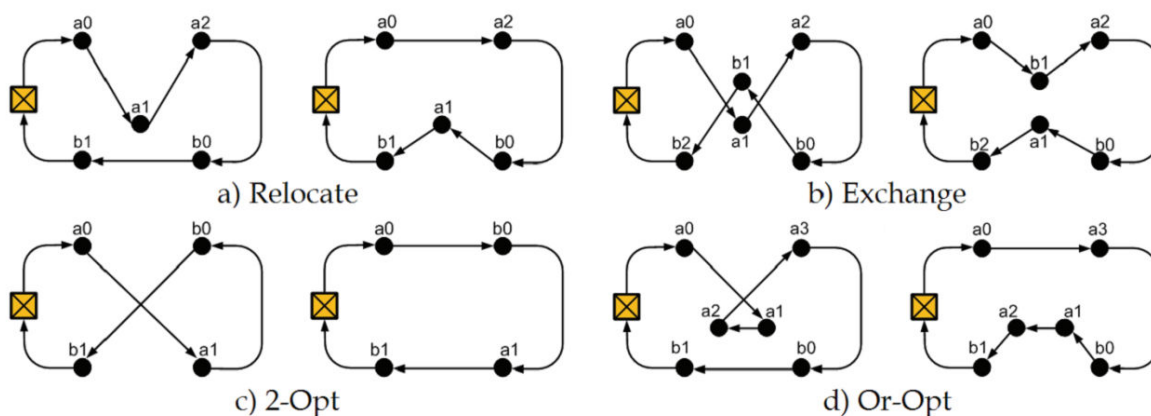
As heurísticas construtivas constroem gradualmente uma solução possível, obedecendo às restrições de custo da solução, mas não contêm uma fase de melhoria em si, isso pode fazer a solução ficar presa a mínimos locais. Nas heurísticas de duas fases, o problema é decomposto nos seus dois componentes naturais: agrupamento dos vértices em rotas possíveis e

construção da rota ótima atual, com a possível repetição destes processos. Esse grupo se subdivide em duas classes: *route-first*, *second-cluster* e *cluster-first*, *route-second*.

Na classe do tipo *route-first*, *second-cluster*, primeiro é feito o processo de definição de rotas para na sequência ser feito o processo de geração de *cluster*, que são os agrupamentos de vértices. Já nos casos de *cluster-first*, *route-second*, primeiro é feito o processo de geração de cluster para depois serem definidas as rotas. Finalmente, os métodos de melhoria tentam aperfeiçoar as soluções possíveis através de uma sequência de trocas de vértices ou ligações entre ou dentro das rotas. Dentro desse grupo também existe uma subdivisão: melhoria intrarrotas e melhoria inter-rotas.

Na melhoria intrarrotas o processo é feito entre os vértices já definidos na rota ótima, apenas trocando-os de posição. Já na melhoria inter-rotas, além dos vértices locais, são explorados vértices vizinhos, pertencentes a outras rotas. Esse processo de melhorias intrarrotas e inter-rotas foi detalhado por Carić et al. (2008) e pode ser visualizado de maneira mais clara nas Figuras 5 e 6, respectivamente.

Figura 5 - Movimentos de melhoria intrarrotas

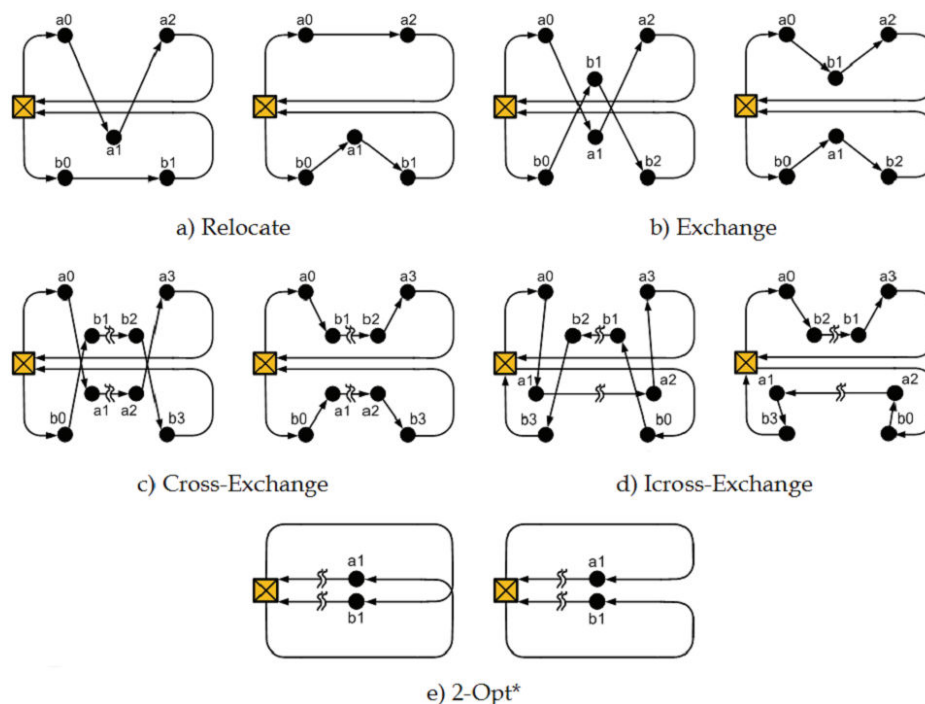


Fonte: Carić et al (2008)

Os movimentos intrarrotas podem ser realizados a partir da realocação da disposição das arestas (a), da troca de alguns vértices que forma a rota (b) ou a partir da troca entre pares de vértices (c) ou ainda entre trios de vértices, quartetos e assim por diante.



Figura 6 - Movimento de melhoria inter-rotas



Fonte: Carić et al (2008)

Nos casos de movimentos inter-rotas, são aplicados os mesmos movimentos descritos para as trocas intrarrotas, com a diferença de, neste caso, existir a troca entre rotas adjacentes, explorando o espaço de vizinhança para realizar esses movimentos.

### 3.1.1 Heurística de Clark e Wright

Essa heurística foi proposta por Clark e Wright (1964) e passou a ser utilizada em problemas de roteamento de veículos pela sua simplicidade e qualidade das soluções geradas. É uma heurística de economias, que caracteriza-se por computar uma lista de economias baseada na visita realizada a todos os vértices do grafo de distribuição. Essa abordagem é capaz de resolver problemas de roteamento de veículos de diferentes classes, além de poder ser associada a outros algoritmos para melhorar seu desempenho (QUEIROZ, 2012).

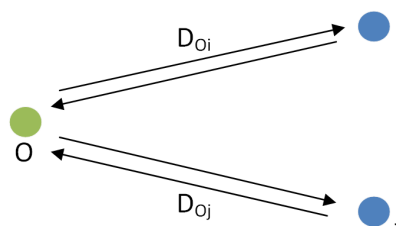
O processo de funcionamento da heurística baseia-se na economia em realizar determinados trajetos a partir da origem, visitando todos os pontos associado ao problema. Tomando a existência de  $n$  pontos (clientes) a serem visitados partindo da origem  $O$  e retornando a mesma após realizar a visita. A

princípio admite-se a pior solução, que seria a alocação de  $n$  veículos para realizar as entregas. Cada veículo parte da origem, visita um ponto e retorna para a origem. Faz isso para cada um dos pontos a ser atendido, conforme Figura 7, onde a distância total é dada por:

$$d = 2 * (d_{oi} + d_{oj}),$$

onde  $d$  é a distância total,  $d_{oi}$  a distância da origem ao ponto  $i$  e  $d_{oj}$  a distância da origem até o ponto  $j$ .

Figura 7 - Primeira etapa do processo de economias de *Clark e Wright*



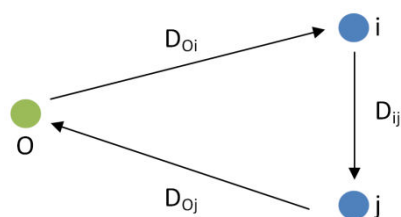
Fonte: O autor

O próximo passo é eliminar um dos veículos envolvidos na entrega, de modo que só um veículo percorra os três pontos. Assim, o veículo realizará o percurso  $Oij$ , retornando a  $O$  no final do percurso, conforme a Figura 8. Dessa forma há uma economia da distância percorrida, pois em vez de fazer quatro trajetos ( $Oi$ ,  $iO$ ,  $Oj$ ,  $jO$ ) ele faz apenas 3 trajetos ( $Oi$ ,  $ij$ ,  $jO$ ). Com isso, a economia gerada nesse novo trajeto pode ser representada por:

$$S_{ij} = d_{oi} + d_{oj} - d_{ij},$$

onde  $S_{ij}$  é a soma das distâncias computadas,  $d_{oi}$  a distância da origem até o ponto  $i$ ,  $d_{oj}$  a distância da origem até o ponto  $j$  e  $d_{ij}$  a distância entre os pontos  $i$  e  $j$ .

Figura 8 - Segunda etapa do processo de economias de *Clark e Wright* (1962).



Fonte: O autor

É esse processo de economias que é empregado na heurística de *Clark e Wright* (1962), onde para cada ponto de um conjunto de  $n$  pontos é computada a lista de economias em ordem decrescente e finalmente são montados os roteiros, executando os seguintes passos:

**Passo 1:** Calcular a lista de economias  $S_{ij} = c_{10} + c_{0j} - c_{ij}$  para  $i, j = 1, \dots, n$ , com  $i \neq j$ . Após obter a lista  $S_{ij}$ , ordená-la em ordem decrescente.

**Passo 2:** Processar a lista de economias começando pelo valor mais alto. Ao analisar as economias, incluir em uma rota obedecendo às restrições existentes no problema. Nesse caso, segundo Caccetta (2013), os seguintes casos precisam ser considerados:

- **Caso 1:** Se nem  $i$  ou  $j$  foram adicionados a uma rota, uma nova rota é iniciada incluindo ambos os nós.
- **Caso 2:** Se inicialmente  $i$  ou  $j$  forem incluídos em uma rota existente em que o ponto anterior não seja adjacente, o mesmo poderá ser adicionado a essa mesma rota, a menos que viole a restrição de capacidade existente no problema.
- **Caso 3:** Se  $i$  e  $j$  já forem incluídos em duas rotas existentes, essas rotas poderão ser incorporadas, desde que não violem as restrições do problema.

**Passo 3:** Se a lista de economias  $S_{ij}$  não se esgotou, retorne ao passo 2, caso contrário, o processo está concluído.

### 3.1.2 Heurística de Gillet e Miller (sweep)

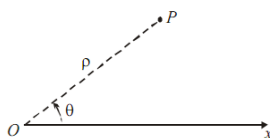
Introduzido na literatura por Gillet e Miller (1974) o algoritmo de varredura (*sweep algorithm*) foi aplicado na resolução de um problema de

roteirização de veículos com frota homogênea com o objetivo de determinar os roteiros de entrega que minimizam a distância total percorrida, de forma que as restrições de capacidade de veículo e distância máxima de cada veículo sejam respeitadas (BELFIORE, 2006).

O algoritmo de varredura é classificado como uma heurística de duas fases (*cluster-first, route-second*), que caracteriza-se na decomposição do problema em dois componentes principais: clusterização e roteamento. Na fase de clusterização, os vértices são divididos em grupos de acordo com características específicas com o objetivo de gerar rotas viáveis a partir de um agrupamento baseado em vizinhança. Na segunda fase é onde ocorre a construção das rotas dentro de cada cluster, determinando o menor caminho em cada agrupamento.

Para determinar os agrupamentos de pontos de entrega, o algoritmo de varredura usa o conceito de coordenadas polares. Um sistema de coordenadas polares no plano consiste em um ponto  $O$ , denominado origem, e de uma semirreta  $OA$ , com origem em  $O$ , denominada eixo polar. O algoritmo supracitado usa esse princípio para realizar a primeira etapa da resolução do problema, fixando o depósito como origem e utilizando um ponto escolhido aleatoriamente como o ponto inicial e geração da semirreta. A partir do ponto inicial, a semirreta gerada faz um movimento anti-horário, agrupando os pontos de distribuição de acordo com as restrições do problema de roteamento, nesse caso, a capacidade dos veículos. Na Figura 9, pode-se ver a demonstração gráfica de geração da semirreta.

Figura 9 - Semirreta gerada com base no sistema de coordenadas polares



Fonte: O autor

Ao aplicar o método de geração da semirreta ao roteamento de veículos, consideramos o plano como sendo o mapa de propriedades a serem atendidas e a origem como sendo o depósito central. É selecionado um dos pontos de distribuição arbitrariamente para definir a semirreta e, a partir dela, iniciar o

processo de geração de grupos, de acordo com a capacidade dos veículos. O movimento de varredura pode acontecer para frente (*forward sweep*), onde a varredura inicia a partir da origem no sentido anti-horário, e a varredura para trás (*backward sweep*), que é executada no sentido horário. Para calcular o ângulo de todas as coordenadas polares é utilizada a seguinte equação:

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

Para complementar a explicação sobre o procedimento desta heurística, uma versão simplificada da heurística é descrita pelos passos a seguir Sosa (2007), onde cada cliente  $i$ , é dado por suas coordenadas polares  $(p_i, \theta_i)$ :

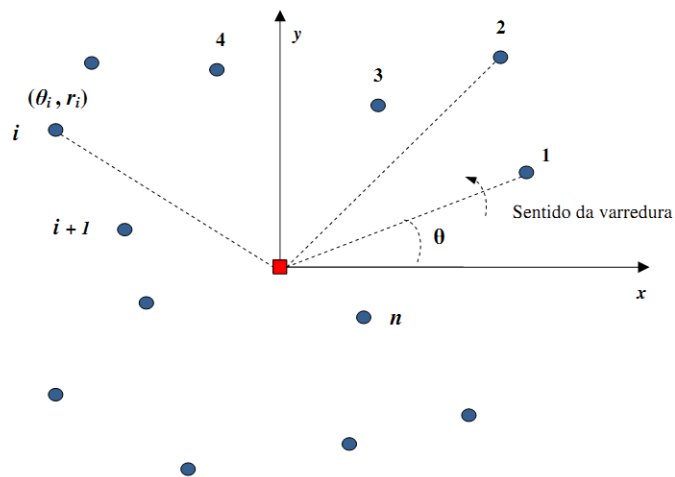
**Passo 1:** Ordenar os clientes segundo o ângulo polar  $\theta$  que cada cliente faz com a semirreta. Se dois clientes possuem o mesmo valor de  $\theta$ , o cliente com menor valor de  $p$  é selecionado. Selecione um cliente  $u_i$  como cliente inicial e faça  $i = 1, k=1; C_1 = \{u_i\}$  é o cluster de ordem 1.

**Passo 2:** Se todos os clientes pertencem a algum cluster, ir ao Passo 4. Caso contrário, ir ao Passo 3.

**Passo 3:** Fazer  $i = i + 1$  e selecionar o próximo cliente  $u_i$ . Se  $u_i$  puder ser adicionado a  $C_k$ , fazer  $C_k = C_k \cup \{u_i\}$ . Caso contrário, fazer  $k = k+1$  e criar um novo agrupamento  $C_k = \{u_i\}$ . Ir ao Passo 2.

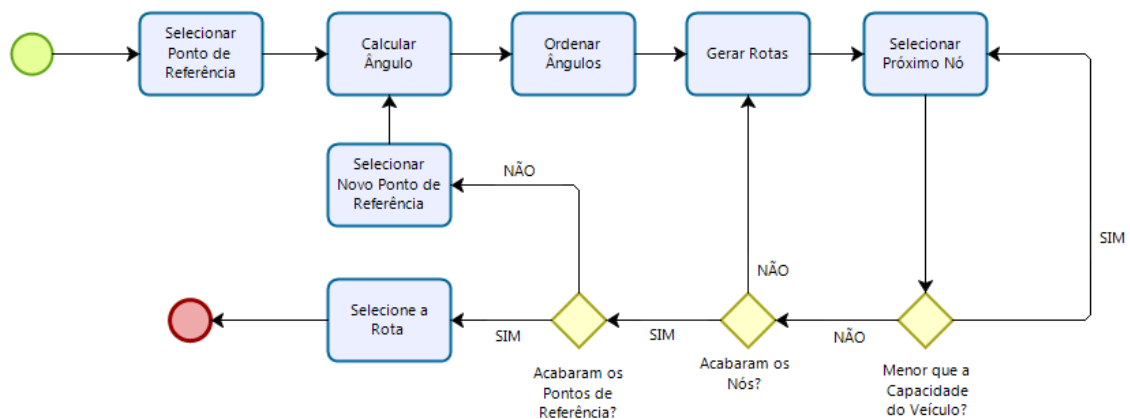
**Passo 4:** Para cada agrupamento  $C_k$ , resolver um Problema do Caixeiro Viajante.

O processo descrito é ilustrado de acordo com a Figura 10.

Figura 10 - Processo de *clusterização*

Fonte: O autor

Essa heurística foi utilizada por Liao e Hu (2011) como parte do processo de geração de soluções utilizado em seu trabalho, onde as etapas do algoritmo foram representadas pelo fluxograma apresentado na Figura 11.

Figura 11 - Fluxograma do algoritmo *Sweep*.

Fonte: O Autor, adaptado de Liao e Hu (2011).

O algoritmo possui a vantagem de ter uma lógica relativamente simples, o que facilita sua implementação, além de possuir a característica de dividir o problema em regiões com pontos adjacentes, o que pode ser relevante em problemas reais de distribuição de produtos.

## 3.2 META-HEURÍSTICAS

O termo meta-heurística, introduzido por Glover (1986) deriva da composição de duas palavras gregas, heurística (do verbo *heuristiké*) que significa “arte de encontrar, descobrir”, e o prefixo meta (*metá*) que exprime a idéia de “nível superior, maior generalidade” (CHAVES, 2009).

Uma meta-heurística é formalmente definida como um processo de geração iterativa, que guia a heurística subordinada através da combinação de diferentes conceitos inteligentes, pela exploração do espaço de pesquisa, sendo que estratégias de aprendizagem são utilizadas para estruturar a informação de modo a encontrar eficientemente soluções quase ótimas (OSMAN e LAPORTE, 1996).

Uma das características das meta-heurísticas é que elas tentam fugir de locais ótimos, o que garante uma maior exploração do espaço de soluções. Essas soluções exploram um espaço de vizinhança entre a solução ótima encontrada até então, buscando realizar melhorias a partir da troca de um ou mais vértices que resulte em um melhor resultado.

Segundo Laporte (2009) as meta-heurísticas podem ser classificadas em busca local, busca populacional ou mecanismos de aprendizagem. Nas meta-heurísticas de busca local, o espaço de soluções é explorado movendo a cada iteração da solução atual para outra solução localizada em sua vizinhança. Nas meta-heurísticas de busca populacional, trabalha-se com populações de soluções, que são recombinações para gerar soluções melhoradas e consideradas superiores. Por fim, as meta-heurísticas baseadas em mecanismos de aprendizagem incluem redes neurais, que tem a capacidade de a cada iteração aprender com a sua experiência e ajustar automaticamente seus pesos (CORDEAU et al, 2005).

### 3.2.1 *Tabu Search*

A meta-heurística *Tabu Search* foi introduzida na literatura por Glover (1986) e caracteriza-se por ser uma meta-heurística de busca local que passou a ser utilizada diante da sua eficiência na resolução de PRV por ser considerada por Cordeau et al. (2002) como a melhor meta-heurística para a

resolução de PRV. Aplicações desse método podem ser vistas em Osman (1993); Gendreau, Hertz e Laporte (1994); Gomes (2011) e Gendreau (1999).

O estado da arte dos métodos de resolução de PRV indica que meta-heurísticas, especialmente a *Tabu Search*, tem tempo de execução aceitável quando comparado com outras abordagens, produzindo soluções de alta qualidade em tempo computacional razoável (MONTANÉ, 2006). Nesse caso, soluções de alta qualidade, são entendidas como soluções que tem uma aproximação de otimização muito próximas às soluções que seriam obtidas por métodos exatos, sendo que essa expressão é comumente usada nos casos de aproximação aos resultados tidos como ótimos.

A meta-heurística *Tabu Search* é um método de busca local que consiste em explorar o espaço de soluções, movendo-se de uma solução para outra que seja seu melhor vizinho. Esta estratégia, juntamente com uma estrutura de memória para armazenar as soluções geradas, permite que a busca não fique presa em um ótimo local (SOUZA, 2011).

Pode-se comprovar a eficiência desse método em Montané (2006), onde é descrito o desenvolvimento de um algoritmo de *Tabu Search* com movimentos de intensificação e diversificação que foram testados em 87 problemas de teste, variando entre 50 a 440 clientes. Em todos os testes realizados houve a melhoria nos resultados propostos.

Uma técnica de busca local tem como característica partir de uma solução inicial e se mover nos espaços de solução em sua vizinhança. Para isso é preciso definir os movimentos para explorar esse espaço de soluções. Além disso, é necessário estipular um critério de parada. O processo de determinação de soluções dessa meta-heurística foi descrito por Glover (1998); Glover (1990) e Laguna (1994), sendo demonstrado em pseudocódigo na Figura 12, onde é detalhado o procedimento desse método.



Figura 12 - Algoritmo de definição da meta-heurística *Tabu Search*.**Algoritmo 1** Busca Tabu

---

```

1: Entrada:  $f(\cdot), N(\cdot), |T|, |V|, s$ 
2: Saída:  $s^*$ 
3:  $s^* \leftarrow s$            {Melhor solução encontrada até então}
4:  $Iter \leftarrow 0$        {Contador do número de iterações}
5:  $MelhorIter \leftarrow 0$  {Iteração mais recente que forneceu  $s^*$ }
6:  $T \leftarrow \emptyset$    {lista tabu}
7: enquanto Critério de parada não satisfeito faça
8:    $Iter \leftarrow Iter + 1$ 
9:   Seja  $s' \leftarrow s \oplus m$  o melhor elemento de  $V \subseteq N(s)$  tal que o movimento  $m$  não seja tabu ( $m \notin T$ )
     ou  $s'$  atenda a um critério de aspiração
10:  Atualize a lista tabu  $T$ 
11:   $s \leftarrow s'$ 
12:  se  $f(s) < f(s^*)$  então
13:     $s^* \leftarrow s$ 
14:     $MelhorIter \leftarrow Iter$ 
15:  fim se
16: fim enquanto
17: Retorne  $s^*$ 

```

---

Fonte: Adaptado de Laguna (1994)

Esse algoritmo recebe como entrada uma solução inicial definida de forma a permitir que, durante a execução do algoritmo, ele possa explorar o espaço de soluções e realizar a melhora, fornecendo como saída uma solução melhor que a solução especificada inicialmente. O processo de determinação inicial pode ser fornecido por uma heurística construtiva, que pode ser baseado em vizinhança, ou outros métodos, como o uso uma heurística em duas fases (BRANDÃO, 2009).

## 4 MATERIAIS E MÉTODOS

### 4.1 ALGORITMOS DESENVOLVIDOS

Foram desenvolvidas duas heurísticas e uma meta-heurística, a partir da análise de métodos de resolução apresentados por Laporte (1992), que cita os resultados satisfatórios no uso de heurísticas e os resultados promissores em pesquisas realizadas utilizando meta-heurísticas. O primeiro método de resolução desenvolvido foi uma heurística construtiva, descrita por Clark e Wright (1962) e o segundo, uma heurística em duas fases, a heurística de varredura (*sweep*) descrito por Gillet e Miller (1974). A meta-heurística desenvolvida foi a Busca Tabu (*Tabu Search*), seguindo definições de Glover (1989) e Glover (1990).

A heurística de *Clark e Wright* apresenta boas soluções para PRV, além de suportar e ser mais flexível nos casos onde se faz necessária a adição de restrições, como a carga máxima dos caminhões ou janelas de tempo (BELFIORE, 2006). A heurística *Sweep* foi desenvolvida para a resolução PRV com frotas homogêneas e pode ser empregada de maneira isolada como método de resolução, como em Renaud, Boctor e Laporte (1996) ou ainda como um método gerador de soluções iniciais para meta-heurísticas, como descrito em Gandelman (2010). Nesse trabalho ela é usada de maneira isolada como um dos métodos disponíveis para resolução de PRV.

A meta-heurística escolhida para fazer parte do sistema de apoio a decisão neste trabalho foi a *Tabu Search* (GLOVER, 1989), por apresentar bons resultados quando aplicada a PRV (GOMES, 2011). A adição de uma meta-heurística foi baseada no fato da mesma explorar de maneira mais ampla o espaço de solução para PRV. A partir de uma solução inicial, esse método explora todo o espaço de vizinhança da solução inicial, buscando novos resultados e fugindo assim de ótimos locais (CORDEAU et al, 2012).

Todos os algoritmos deste trabalho foram desenvolvidos utilizando a linguagem de programação Java (JAVA, 2014), tendo com ferramenta de desenvolvimento a IDE Eclipse Juno (ECLIPSE, 2014), versão 22.0.5-757759.

## 4.2 BASES DE DADOS

Nesse trabalho foram utilizados dados provenientes de três bases, de onde foram selecionados arquivos de texto, denominados instâncias, que contém registros de problemas de roteamento de veículos com frotas homogêneas e foram utilizados como entrada para validação e testes de desempenho dos algoritmos desenvolvidos. Essas bases são disponibilizadas em sites específicos e agrupam instâncias de problemas de roteamento de veículos resolvidos por diferentes pesquisadores em seus trabalhos. Essas instâncias nada mais são do que dados referentes a problemas de roteamento de veículos, que trazem dados como número de pontos a serem atendidos, as coordenadas desses pontos (utilizadas para os cálculos de distância), sua demanda, além da capacidade dos caminhões utilizados para a distribuição.

Foram utilizadas as instâncias do site da *Computational Infrastructure for Operations Research* (COIN, 2014), cujo objetivo é fornecer recursos para o desenvolvimento de pesquisas na área de otimização combinatória, do site *Networking and Emerging Optimization* (NEO, 2014) dedicado ao estudo dos PRV e suas variações e também o site do professor e autor de vários trabalhos sobre PRV, *Marius M. Solomon* (SOLOMON, 2014), que traz detalhes dos trabalhos desenvolvidos por esse pesquisador, juntamente com os dados sobre problemas de PRV resolvidos por ele.

As instâncias de problemas foram selecionadas levando-se em consideração a característica dos problemas, buscando-se escolher instâncias com dados similares ao problema estudado nesse trabalho, ou seja, problemas de roteamento de veículos com frotas homogêneas, onde a principal restrição é a capacidade dos veículos. Nos testes dessas instâncias, a variável de otimização utilizada foi a distância entre os pontos.

### **Instâncias de Augerat et al.**

As instâncias desse grupo foram publicadas por Augerat et al. (1995) e são divididas em três conjuntos distintos: conjunto A, conjunto B e conjunto P. O conjunto A é formado por 27 instâncias, o conjunto B por 23 e o conjunto P por 24. Todas são instâncias para VRP com frota homogênea. Para este trabalho foram escolhidas 12 instâncias entre os três conjuntos disponíveis. As

instâncias foram selecionadas de acordo com o número de pontos necessários para o atendimento, com tamanhos pequenos, médios e grandes. Esses dados são detalhados no Quadro 1.

Quadro 1 - Detalhamento das instâncias de Augerat et al. (1995)

| Instância | Número de Propriedades | Carga dos Veículos | Resultado Ótimo |
|-----------|------------------------|--------------------|-----------------|
| A-n32-k5  | 32                     | 100                | 784             |
| A-n44-k6  | 44                     | 100                | 937             |
| A-n60-k9  | 60                     | 100                | 1408            |
| A-n80-k10 | 80                     | 100                | 1764            |
| B-n31-k5  | 31                     | 100                | 672             |
| B-n45-k5  | 45                     | 100                | 751             |
| B-n63-k10 | 63                     | 100                | 1537            |
| B-n78-k10 | 78                     | 100                | 1266            |
| P-n16-k8  | 16                     | 100                | 435             |
| P-n50-k7  | 50                     | 150                | 554             |
| P-n70-k10 | 70                     | 135                | 834             |
| P-n101-k4 | 101                    | 400                | 681             |

Fonte: Adaptado de Augerat et al. (1995).

### **Instâncias de Christofides e Eilon**

Esse grupo de instâncias foi publicado em Christofides e Eilon (1969). É formado por apenas um grupo de instâncias, conhecida como conjunto E, composto por 15 instâncias com números de consumidores variando entre 13 e 101 e com cargas de veículos distintas. Ambas são instâncias para VRP com frota homogênea. Foram escolhidas instância com variação gradativa de tamanho, com dimensões pequenas, médias e grandes. As instâncias são detalhadas no Quadro 2.

Quadro 2 - Detalhamento das instâncias de Christofides e Eilon (1969).

| Instância  | Número de Propriedades | Carga dos Veículos | Resultado Ótimo |
|------------|------------------------|--------------------|-----------------|
| E-n22-k4   | 22                     | 6000               | 375             |
| E-n23-k3   | 23                     | 4500               | 569             |
| E-n30-k3   | 30                     | 4500               | 534             |
| E-n33-k4   | 33                     | 8000               | 835             |
| E-n51-k5   | 51                     | 160                | 521             |
| E-n76-k7   | 76                     | 220                | 683             |
| E-n76-k8   | 76                     | 180                | 735             |
| E-n76-k10  | 76                     | 140                | 832             |
| E-n76-k14  | 76                     | 100                | 1032            |
| E-n101-k8  | 101                    | 200                | 817             |
| E-n101-k14 | 101                    | 112                | 1077            |

Fonte: Adaptado de Christofides e Eilon (1969)

### Instâncias de Fisher

As instâncias desse grupo são atribuídas a Fisher (1994); Fisher e Jaikumar (1981), que criaram três instâncias com números variados de consumidores e com cargas de veículos distintas. Ambas são instâncias para PRV com frota homogênea. As instâncias são detalhadas no Quadro 3.

Quadro 3 - Detalhamento das Instâncias de Fisher (1994)

| Instância | Número de Propriedades | Carga dos Veículos | Resultado Ótimo |
|-----------|------------------------|--------------------|-----------------|
| F-n45-k4  | 45                     | 2010               | 728             |
| F-n72-k4  | 72                     | 30000              | 237             |
| F-n135-k7 | 135                    | 2210               | 1162            |

Fonte: Adaptado de Fisher (1994)

### 4.3 TESTES COMPUTACIONAIS

O cenário utilizado para os testes tem como base de dados as instâncias para problemas de roteamento de veículos disponíveis em Coin (2014), Neo (2014) e Solomon (2014), que são os dados de entrada usados nos algoritmos heurísticos desenvolvidos neste trabalho. O cenário caracteriza-se pela utilização da frota de caminhões homogênea, entregas realizadas a partir de um único depósito e demandas de entrega localizadas em cada cliente. A variável de otimização utilizada é a distância entre os pontos e os algoritmos

devem buscar a menor distância ao percorrê-los para fazer a entrega, respeitando as restrições de capacidade dos caminhões.

As instâncias possuem regras de estruturação definidas em Reinelt (2014) que descreve o padrão para o registro de problemas de roteamento de veículos de acordo com as características que são apresentadas. Os arquivos trazem a informação com o nome da instância, os autores do trabalho onde as instâncias foram inicialmente propostas, o resultado ótimo conhecido até então e a capacidade do veículo utilizado. Também são descritos os pontos com sua respectiva coordenada e demanda. O formato desses arquivos é mostrado na Figura 13.

Figura 13 - Padrão de arquivos para PRV

```

NAME : E-n22-k4
COMMENT : (Christophides and Eilon, Min no of trucks: 4, Optimal value: 375)
TYPE : CVRP
DIMENSION : 22
EDGE_WEIGHT_TYPE : EUC_2D
CAPACITY : 6000
NODE_COORD_SECTION
1 145 215
2 151 264
3 159 261
.
. /*Dados suprimidos*/
.
20 129 189
21 155 185
22 139 182
DEMAND_SECTION
1 0
2 1100
3 700
.
. /*Dados suprimidos*/
.
20 2500
21 1800
22 700
DEPOT_SECTION
1
-1
EOF

```

Fonte: O autor

O sistema de validação dos algoritmos é responsável por carregar os dados dos arquivos, instanciar as classes necessários para determinar a matriz de adjacências com os pontos definidos no arquivo e executar os algoritmos heurísticos desenvolvidos nesse trabalho. Com isso é possível determinar as soluções que cada algoritmo gera para a mesma instância, a comparação dos resultados dos diferentes algoritmos, além da comparação com os melhores resultados encontrados até então para as instâncias arquivadas. Visto que as bases utilizadas como entrada do problema são de domínio público e servem como base de dados para trabalhos que buscam melhorar a solução dessas

instâncias, assim como em Tang, Zhang e Pan (2010); Martinhon, Lucena e Maculan (2004); Repoussis et al (2010) o seu uso torna-se justificado.

#### 4.3.1 Cálculo Para Distâncias Euclidianas

Para o cálculo de distância entre os pontos de entrega, assim como em outros trabalhos que utilizam as instâncias de problemas de roteamento de veículos, como Tang, Zhang e Pan (2010); Martinhon, Lucena e Maculan (2004); Repoussis et al (2010) foi utilizado o cálculo distâncias euclidianas relativas às coordenadas dos pontos. A distância euclidiana consiste no cálculo da distância entre dois pontos quaisquer, ligados por uma reta, tal como o Teorema de Pitágoras. Para esse cálculo é utilizada a fórmula descrita na Equação 1, que utilizada as coordenadas  $x$  e  $y$  de um ponto no plano para determinar a distância entre os pontos.

$$d_{ij} = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2}, (1)$$

onde  $i$  é o ponto de origem e  $j$  é o ponto de destino. Baseado em distâncias euclidianas que utiliza como cálculo essa fórmula, é possível determinar a matriz de adjacências, que armazena a distância entre os pontos.

É a partir do cálculo das distâncias que é feita a montagem do grafo de distribuição referente aos pontos de entrega, com suas respectivas demandas. Os vértices do grafo representam as propriedades e as arestas a sua ligação. Nos arcos do grafo são adicionadas as variáveis de decisão na escolha do caminho otimizado, que nesse caso é a distância percorrida.

#### 4.3.2 Cálculo Para Distâncias Baseadas em Coordenadas Geográficas

Como nesse trabalho estuda-se também a aplicação dos algoritmos em um simulador que possa ser utilizado em ambientes reais de roteamento de veículos, é necessário realizar o cálculo de distância para pontos determinados por coordenadas geográficas. Nesse caso, usa-se a distância linear da trigonometria esférica, pois deve-se levar em consideração que são coordenadas do globo terrestre, que possui o formato de uma esfera. Esse cálculo é obtido conforme a Equação 2, definida por Ballou (2001).

$$D_{AB} = 6340 * \{arccos[ \sin(lat_B) * \sin(lat_A) + \cos(lat_A) * \cos(lat_B) * \cos(|long_B - long_A|)]\} \quad (2)$$

onde,

$D_{AB}$  = distância entre os pontos A e B

$lat_A$  = latitude do ponto A

$long_A$  = longitude do ponto A

$lat_B$  = latitude do ponto B

$long_B$  = longitude do ponto B

Nesse caso, o cálculo foi auxiliado pelo uso da biblioteca GeoUtils, que implementa o cálculo a partir de um par de coordenadas geográficas qualquer, sendo possível determinar a matriz de adjacências contendo a distância entre os pontos.



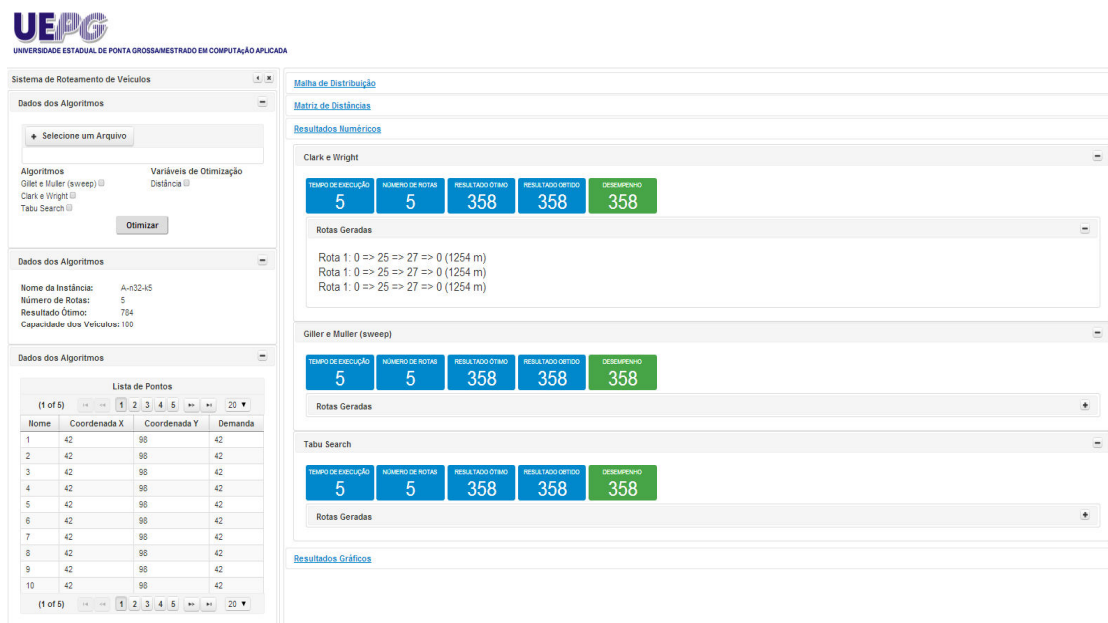
## 5 RESULTADOS

### 5.1 VALIDAÇÃO DOS ALGORITMOS DESENVOLVIDOS

A primeira etapa da construção do sistema de apoio a decisão foi a validação dos algoritmos desenvolvidos. Esses algoritmos compõem o pacote de classes responsáveis pela otimização dos problemas de roteamento de veículos. Essas classes foram modeladas utilizando padrões de entrada e saída para os algoritmos, de modo que seu uso pudesse se estender de maneira modular em diferentes aplicações. Além disso, a estrutura dos dados de saída permite que os mesmos possam ser utilizados para a construção da malha de distribuição disponível na camada SIG desenvolvida nesse trabalho.

Para os testes de validação, foi construído um sistema conforme a Figura 14. Ele tem como entrada dados de instâncias de problemas de roteamento de veículos com frota heterogênea, que obedecem a restrições de carga dos caminhões. A partir desses dados, que incluem informações com as coordenadas dos pontos que precisam ser atendidos, capacidades dos veículos e demanda de cada ponto, o sistema é capaz de realizar a definição de rotas usando os algoritmos selecionados para o roteamento.

Figura 14 - Visão geral do sistema de validação de algoritmos heurísticos



**UEPG**  
UNIVERSIDADE ESTADUAL DE PONTA GROSSA - CENTRO DE CIÊNCIAS EXATAS E TECNOLÓGICAS

Sistema de Roteamento de Veículos

Dados dos Algoritmos

Selecione um Arquivo

Algoritmos: Clark e Wright, Tabu Search

Variáveis de Otimização: Distância

Otimizar

Dados dos Algoritmos

Nome da Instância: A-132-15

Número de Rotas: 5

Resultado Ótimo: 784

Capacidade dos Veículos: 100

Dados dos Algoritmos

Lista de Pontos

| Nome | Coordenada X | Coordenada Y | Demanda |
|------|--------------|--------------|---------|
| 1    | 42           | 98           | 42      |
| 2    | 42           | 98           | 42      |
| 3    | 42           | 98           | 42      |
| 4    | 42           | 98           | 42      |
| 5    | 42           | 98           | 42      |
| 6    | 42           | 98           | 42      |
| 7    | 42           | 98           | 42      |
| 8    | 42           | 98           | 42      |
| 9    | 42           | 98           | 42      |
| 10   | 42           | 98           | 42      |

Matriz de Distribuição

Matriz de Distâncias

Resultados Numéricos

Clark e Wright

| TEMPO DE EXECUÇÃO | NÚMERO DE ROTAS | RESULTADO ÓTIMO | RESULTADO DETIDO | DESEMPENHO |
|-------------------|-----------------|-----------------|------------------|------------|
| 5                 | 5               | 358             | 358              | 358        |

Rotas Geradas

Rota 1: 0 => 25 => 27 => 0 (1254 m)

Rota 1: 0 => 25 => 27 => 0 (1254 m)

Rota 1: 0 => 25 => 27 => 0 (1254 m)

Giller e Muller (sweep)

| TEMPO DE EXECUÇÃO | NÚMERO DE ROTAS | RESULTADO ÓTIMO | RESULTADO DETIDO | DESEMPENHO |
|-------------------|-----------------|-----------------|------------------|------------|
| 5                 | 5               | 358             | 358              | 358        |

Rotas Geradas

Tabu Search

| TEMPO DE EXECUÇÃO | NÚMERO DE ROTAS | RESULTADO ÓTIMO | RESULTADO DETIDO | DESEMPENHO |
|-------------------|-----------------|-----------------|------------------|------------|
| 5                 | 5               | 358             | 358              | 358        |

Rotas Geradas

Resultados Gráficos

Fonte: O autor

Os dados são carregados pelo sistema a partir de arquivos de texto com a formatação definida por Reinelt (2014). Após o carregamento dos arquivos são coletados os dados sobre a identificação da instância utilizada, capacidade de carga dos caminhões, o melhor resultado encontrado até o momento para a instância (resultado ótimo) e os dados dos pontos a serem atendidos, juntamente com as coordenadas dos pontos e a demanda de cada um. Além disso, também são mostradas as opções de otimização, como os algoritmos disponíveis e as variáveis de restrição utilizadas.

Figura 15 - Opções de roteamento do sistema

**Sistema de Roteamento de Veículos**

**Dados dos Algoritmos**

Selecionar um Arquivo

Algoritmos: Gillet e Muller (sweep)  Clark e Wright  Tabu Search

Variáveis de Otimização: Distância

Otimizar

**Dados da Instância**

Nome da Instância: A-n32-k5  
 Número de Rotas: 5  
 Resultado Ótimo: 784  
 Capacidade dos Veículos: 100

**Dados dos Algoritmos**

Lista de Pontos (1 of 5)

| Nome | Coordenada X | Coordenada Y | Demanda |
|------|--------------|--------------|---------|
| 1    | 42           | 98           | 42      |
| 2    | 42           | 98           | 42      |
| 3    | 42           | 98           | 42      |
| 4    | 42           | 98           | 42      |
| 5    | 42           | 98           | 42      |
| 6    | 42           | 98           | 42      |
| 7    | 42           | 98           | 42      |
| 8    | 42           | 98           | 42      |
| 9    | 42           | 98           | 42      |
| 10   | 42           | 98           | 42      |

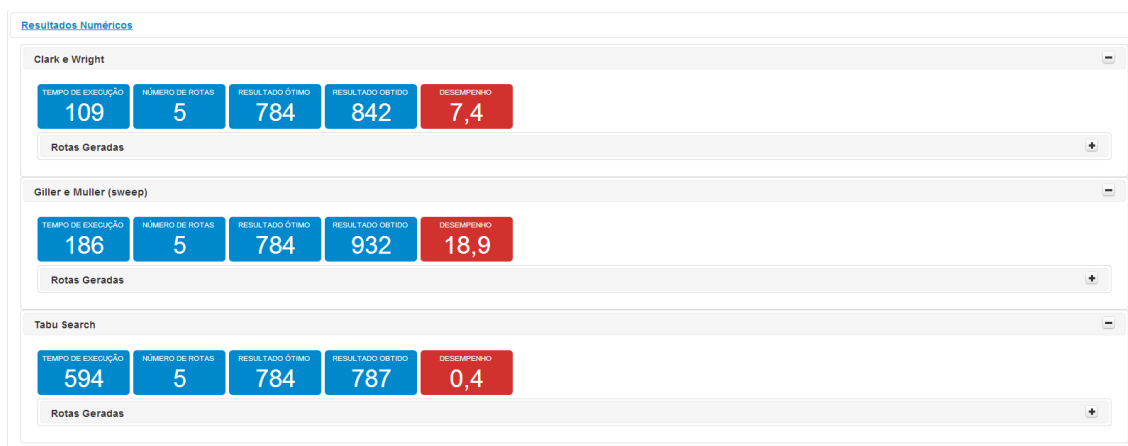
Fonte: O autor

Com esses dados é possível determinar a matriz de adjacências, realizando o cálculo de distância entre os pontos, utilizando a fórmula da distância euclidiana. É essa matriz que compõe o grafo que será percorrido pelos algoritmos de otimização. A matriz de adjacências é uma matriz formada por  $n$  linhas por  $n$  colunas, onde  $n$  é o número de propriedades somado com o ponto de origem de distribuição.

Após o processamento realizado pelos algoritmos, são computados os dados sobre tempo de execução, resultado do total de distância percorrida e o número de rotas especificado. Com isso é possível comparar o resultado do

valor total de distância obtido com o valor ótimo disponível na instância e comparando os dois, calcular o desvio entre a solução ótima e a solução encontrada, gerando o valor percentual que é utilizado como índice de qualidade das soluções. Os dados de saída de resultados dos algoritmos são mostrados conforme Figura 16.

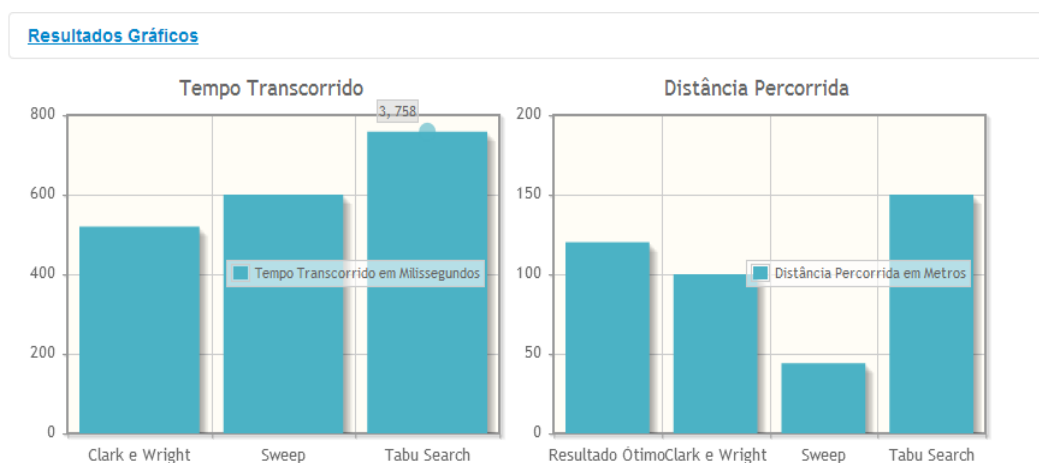
Figura 16 - Tela de exibição de resultados dos algoritmos



Fonte: O autor

Outra ferramenta que auxilia no processo de avaliação dos resultados é a visualização gráfica das soluções, que compara os resultados obtidos e também o tempo de processamento relativo aos três algoritmos desenvolvidos, conforme Figura 17.

Figura 17 - Visualização gráfica dos resultados de otimização



Fonte: O autor

### 5.1.1 Resultados Computacionais

Os resultados computacionais para os testes foram realizados utilizando o sistema para validação de algoritmos. Para a realização dos testes foi utilizado um *notebook Dell<sup>®</sup> Inspiron*, processador *Intel<sup>®</sup> Pentium R T4400*, *dual core*, com 2.20 Ghz, memória RAM de 3 Gb e sistema operacional *Windows<sup>®</sup> 7 Ultimate* de 32 bits.

No primeiro teste foram utilizadas como entradas para o sistema as instâncias de Fisher. Elas foram testadas com os três algoritmos disponíveis. Os resultados são mostrados na Tabela 1.

Tabela 1 - Resultado dos testes computacionais para as instâncias de Fisher

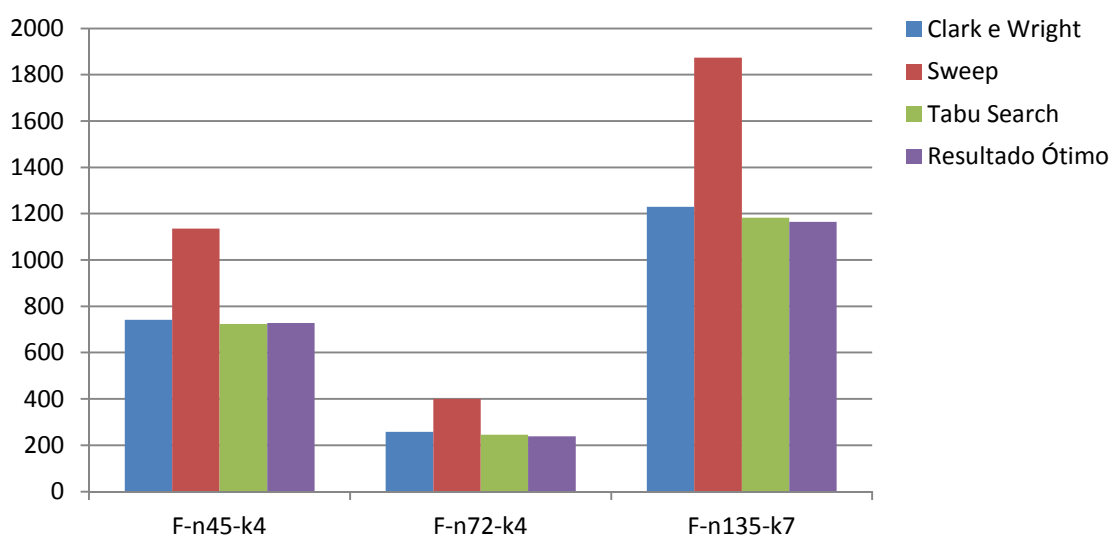
| Instância | Clark e Wright |      |        | Sweep |        | Tabu Search |        |
|-----------|----------------|------|--------|-------|--------|-------------|--------|
|           | SOL            | SO   | DS (%) | SO    | DS (%) | SO          | DS (%) |
| F-n45-k4  | 728            | 742  | 1,9    | 1136  | 56     | 723         | -0,7   |
| F-n72-k4  | 238            | 258  | 8,4    | 400   | 68,1   | 246         | 3,4    |
| F-n135-k7 | 1165           | 1229 | 5,5    | 1873  | 60,8   | 1183        | 1,5    |

Fonte: O autor

Nota: SOL solução ótima disponível na literatura; SO solução ótima encontrada; DS desvio em relação a solução ótima, em porcentagem. Porcentagens positivas tiveram variação acima da SOL e porcentagens negativas tiveram variação abaixo da SOL.

Também é apresentada a representação gráfica dos resultados, conforme Figura 18.

Figura 18 - Gráfico dos resultados para as instâncias de Fisher



Fonte: O autor

Além dos dados de desempenho na determinação de rotas otimizadas, também foram coletados os dados de desempenho computacional, sendo utilizada a variável tempo de execução dos algoritmos. Essa variável registra o tempo necessário para a execução dos algoritmos, em milissegundos. Na Tabela 2, são mostrados os dados de tempo de execução relacionado às instâncias de Fisher, com sua respectiva representação gráfica.

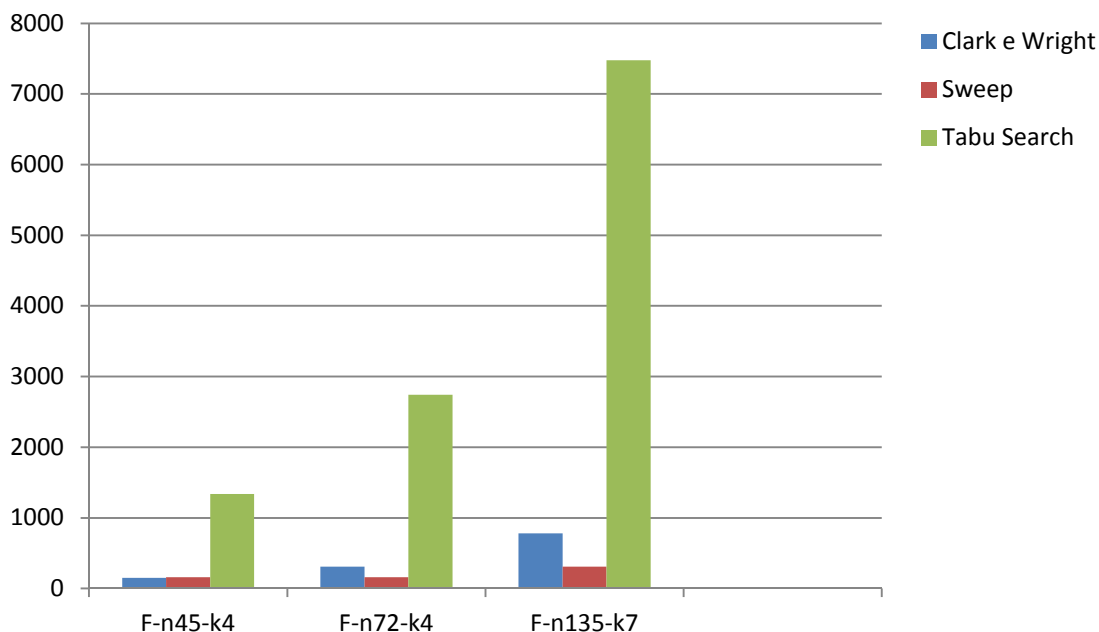
Tabela 2 - Tempo computacional resultante da execução dos algoritmos em milissegundos

|           | <i>Clark e Wright</i> | <i>Sweep</i> | <i>Tabu Search</i> |
|-----------|-----------------------|--------------|--------------------|
| F-n45-k4  | 150                   | 160          | 1340               |
| F-n72-k4  | 310                   | 160          | 2741               |
| F-n135-k7 | 780                   | 310          | 7477               |

Fonte: O autor

Os dados de tempo de processamento são mostrados em gráfico conforme Figura 19.

Figura 19 - Gráfico de tempo para as instâncias de Fisher



Fonte: Ao autor

A partir dos resultados é possível verificar que os melhores estão relacionados à execução da heurística de *Tabu Search*. Em duas das instâncias testadas a solução teve desempenho pior que a disponível na literatura, com desvio percentual baixo de 1,5% e 3,5% enquanto que para a instância F-n45-k4, o resultado foi melhor que a disponível na literatura,

diminuindo seu custo em 0,7%. Pela análise feita, nota-se que o tamanho das amostras não influenciou na qualidade dos resultados.

A heurística de *Clark e Wright* (1962) teve pior desempenho que a *Tabu Search*, porém, com diferença não significativa, com desvios percentuais próximos aos resultados da segunda, não conseguindo ser melhor em nenhuma das instâncias da literatura. A heurística *Sweep* foi a que apresentou pior resultado, muito acima tanto dos resultados ótimos presentes na literatura, quanto dos resultados dos outros métodos desenvolvidos nesse trabalho.

Em relação ao tempo de execução, o algoritmo *Sweep* foi o que teve melhor desempenho em relação aos outros, porém, mantendo certa proximidade com os resultados de *Clark e Wright* (1962). O algoritmo que teve maior tempo computacional e com grande diferença entre os outros foi o *Tabu Search*. Nota-se também que quanto maior o tamanho do conjunto de dados, maior é a discrepância de tempo entre os algoritmos.

Os testes realizados na sequência foram feitos utilizando como entrada para o sistema as instâncias de Christofides e Eilon (1969). Os testes seguiram a mesma metodologia do primeiro, sendo utilizados os três algoritmos disponíveis. Os resultados são mostrados na Tabela 3, com sua respectiva representação gráfica.

Tabela 3 - Resultado dos testes computacionais para as instâncias de Christofides e Eilon (1969).

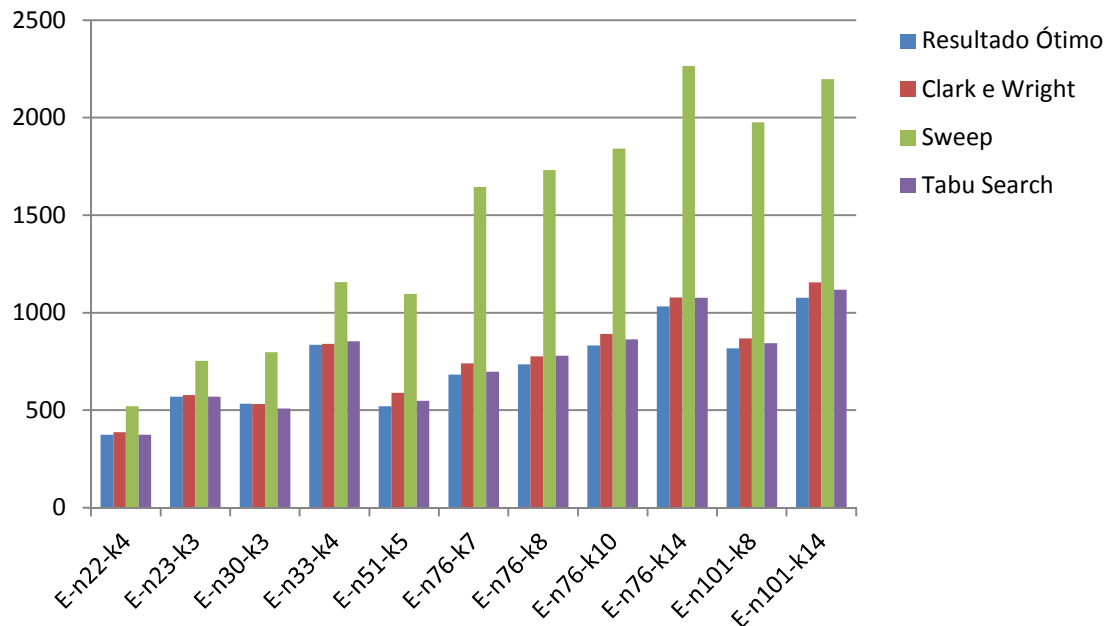
| Instância  | SOT  | Clark e Wright |        | Sweep |        | Tabu Search |        |
|------------|------|----------------|--------|-------|--------|-------------|--------|
|            |      | SO             | DS (%) | SO    | DS (%) | SO          | DS (%) |
| E-n22-k4   | 375  | 387            | 3,2    | 520   | 38,7   | 375         | 0,0    |
| E-n23-k3   | 569  | 577            | 1,4    | 753   | 32,3   | 569         | 0,0    |
| E-n30-k3   | 534  | 532            | -0,4   | 798   | 49,4   | 509         | -4,7   |
| E-n33-k4   | 835  | 841            | 0,7    | 1157  | 38,6   | 853         | 2,2    |
| E-n51-k5   | 521  | 589            | 13,1   | 1097  | 110,6  | 549         | 5,4    |
| E-n76-k7   | 683  | 740            | 8,3    | 1644  | 140,7  | 697         | 2,0    |
| E-n76-k8   | 735  | 777            | 5,7    | 1731  | 135,5  | 779         | 6,0    |
| E-n76-k10  | 832  | 892            | 7,2    | 1841  | 121,3  | 864         | 3,8    |
| E-n76-k14  | 1032 | 1078           | 4,5    | 2265  | 119,5  | 1077        | 4,4    |
| E-n101-k8  | 817  | 869            | 6,4    | 1975  | 141,7  | 843         | 3,2    |
| E-n101-k14 | 1077 | 1155           | 7,2    | 2198  | 104,1  | 1118        | 3,8    |

Fonte: O autor.

Nota: SOL solução ótima disponível na literatura; SO solução ótima encontrada; DS desvio em relação a solução ótima, em porcentagem. Porcentagens positivas tiveram variação acima da SOL e porcentagens negativas tiveram variação abaixo da SOL.

Assim como na execução dos algoritmos mostrados anteriormente, foi elaborado o gráfico de desempenho conforme Figura 20.

Figura 20 - Gráfico dos resultados para as instâncias de Christofides e Eilon (1969).



Fonte: Ao autor

Foram computados os dados de desempenho, mostrados na Tabela 4, com sua respectiva representação gráfica.

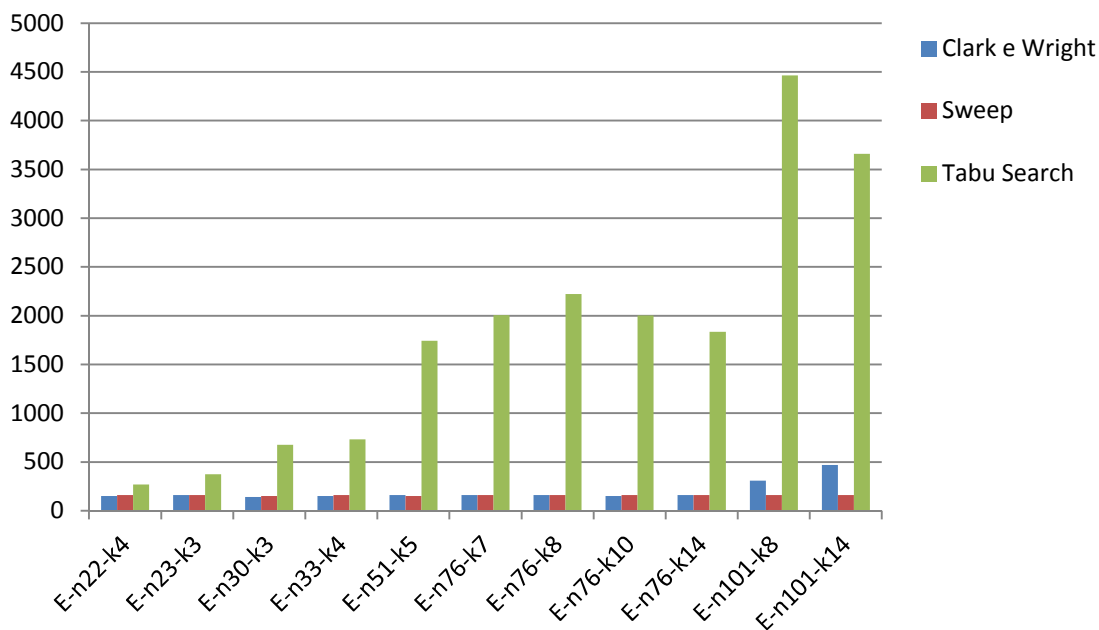
Tabela 4 - Tempo de processamento para os testes computacionais para as instâncias de Christofides e Eilon (1969).

|            | Clark e Wright | Sweep | Tabu Search |
|------------|----------------|-------|-------------|
| E-n22-k4   | 150            | 160   | 269         |
| E-n23-k3   | 160            | 160   | 376         |
| E-n30-k3   | 140            | 150   | 675         |
| E-n33-k4   | 150            | 160   | 732         |
| E-n51-k5   | 160            | 150   | 1742        |
| E-n76-k7   | 160            | 160   | 2004        |
| E-n76-k8   | 160            | 160   | 2221        |
| E-n76-k10  | 150            | 160   | 2002        |
| E-n76-k14  | 160            | 160   | 1835        |
| E-n101-k8  | 310            | 160   | 4464        |
| E-n101-k14 | 470            | 160   | 3661        |

Fonte: O autor.

Os dados são seguidos de sua representação gráfica conforme Figura 21.

Figura 21 - Gráfico de tempo para as instâncias de Christofides e Eilon (1969).



Fonte: Ao autor

Analisando os resultados, mais uma vez nota-se a prevalência do método *Tabu Search* em relação aos outros métodos. Porém, dessa vez, o número de instâncias onde a heurística de *Clark e Wright* apresentou melhor desempenho foi maior que nos testes para as instâncias de Fisher. Isso mostra que as características das instâncias interferem no resultado encontrado.

Em relação aos valores registrados como melhores resultados na literatura, houve a diminuição do tempo pelos métodos *Tabu Search* e de *Clark e Wright* para somente uma instância. Nesse caso, foi a mesma instância E-n30-k3, para ambos os métodos. Novamente a heurística *Sweep* apresentou resultados piores em relação aos outros métodos, com valores discrepantes em relação ao desempenho das soluções.

Quanto aos dados de execução dos algoritmos, as duas heurísticas desenvolvidas nesse trabalho tiveram desempenho similar, com pouca diferença em relação aos resultados. Chama a atenção o tempo gasto pela meta-heurística *Tabu Search*, com valores muito acima dos outros métodos.

Para finalizar foram feitos os testes utilizando como entrada para o sistema as instâncias de Augerat et al. (1995). Os testes seguiram a mesma metodologia dos anteriores, sendo utilizados os três algoritmos disponíveis. Os resultados são mostrados na Tabela 5, com sua respectiva representação gráfica.



Tabela 5 - Resultado dos testes computacionais para as instâncias de Augerat et al. (1995).

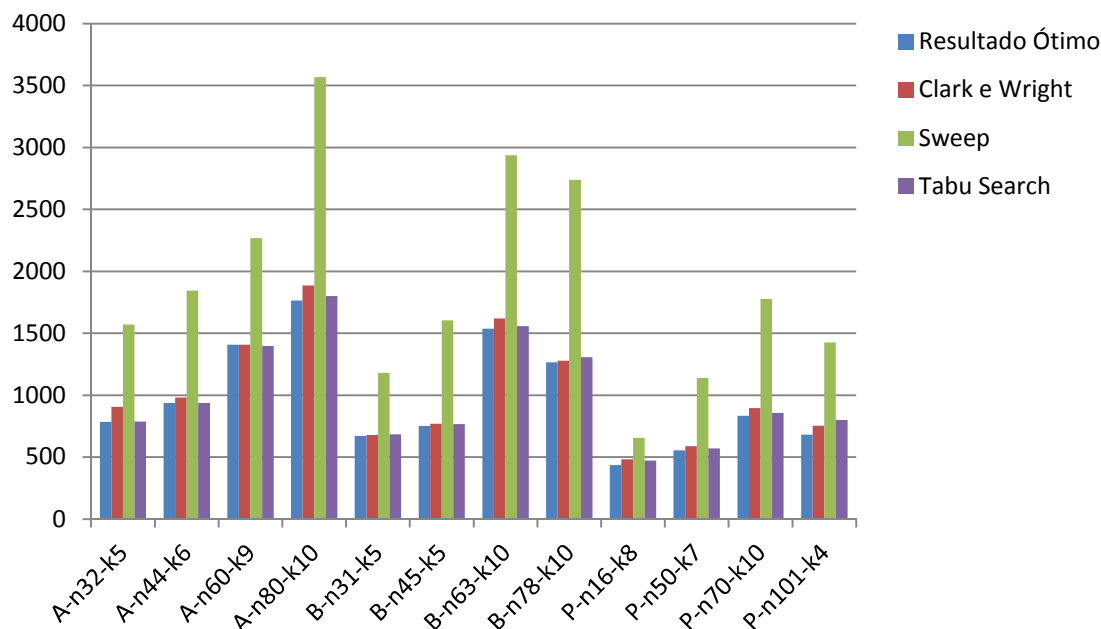
| Instância | Clark e Wright |      |      | Sweep |       | Tabu Search |      |
|-----------|----------------|------|------|-------|-------|-------------|------|
|           | SOT            | SO   | DS   | SO    | DS    | SO          | DS   |
| A-n32-k5  | 784            | 907  | 15,7 | 1570  | 100,3 | 787         | 0,4  |
| A-n44-k7  | 937            | 981  | 4,7  | 1844  | 96,8  | 938         | 0,1  |
| A-n60-k9  | 1408           | 1408 | 0,0  | 2267  | 61,0  | 1398        | -0,7 |
| A-n80-k10 | 1764           | 1885 | 6,9  | 3566  | 102,2 | 1800        | 2,0  |
| B-n31-k5  | 672            | 678  | 0,9  | 1179  | 75,4  | 683         | 1,6  |
| B-n45-k5  | 751            | 770  | 2,5  | 1604  | 113,6 | 766         | 2,0  |
| B-n63-k10 | 1537           | 1620 | 5,4  | 2937  | 91,1  | 1557        | 1,3  |
| B-n78-k10 | 1266           | 1278 | 0,9  | 2738  | 116,3 | 1306        | 3,2  |
| P-n16-k8  | 435            | 482  | 10,8 | 656   | 50,8  | 473         | 8,7  |
| P-n50-k7  | 554            | 588  | 6,1  | 1139  | 105,6 | 569         | 2,7  |
| P-n70-k10 | 834            | 895  | 7,3  | 1776  | 112,9 | 857         | 2,8  |
| P-n101-k4 | 681            | 753  | 10,6 | 1425  | 109,3 | 801         | 17,6 |

Fonte: O autor.

Nota: SOL solução ótima disponível na literatura; SO solução ótima encontrada; DS desvio em relação a solução ótima, em porcentagem. Porcentagens positivas tiveram variação acima da SOL e porcentagens negativas tiveram variação abaixo da SOL.

Os resultados gráficos são apresentados conforme Figura 22.

Figura 22 - Gráfico de resultado para as instâncias de Augerat et al. (1995).



Fonte: Ao autor

Para esses dados, também foi computado o tempo de execução dos algoritmos, conforme Tabela 6.

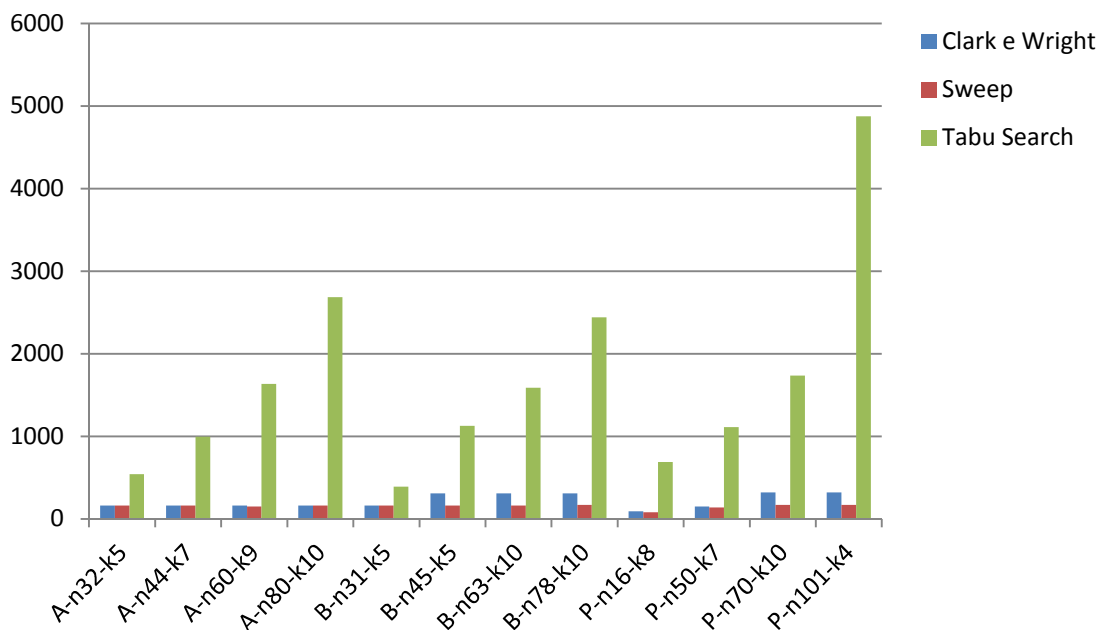
Tabela 6 - Tempo de processamento para os testes computacionais para as instâncias de Augerat et al. (1995).

|           | Clark e Wright | Sweep | Tabu Search |
|-----------|----------------|-------|-------------|
| A-n32-k5  | 160            | 160   | 541         |
| A-n44-k7  | 160            | 160   | 993         |
| A-n60-k9  | 160            | 150   | 1635        |
| A-n80-k10 | 160            | 160   | 2684        |
| B-n31-k5  | 160            | 160   | 392         |
| B-n45-k5  | 310            | 160   | 1128        |
| B-n63-k10 | 310            | 160   | 1587        |
| B-n78-k10 | 310            | 170   | 2441        |
| P-n16-k8  | 90             | 80    | 690         |
| P-n50-k7  | 150            | 140   | 1113        |
| P-n70-k10 | 320            | 170   | 1736        |
| P-n101-k4 | 320            | 170   | 4876        |

Fonte: O autor

Também é apresentado o gráfico com os resultados referentes ao tempo de execução, conforme Figura 23.

Figura 23 - Gráfico de tempo para as instâncias de Augerat et al. (1995).



Fonte: Ao autor

Para as instâncias de Augerat et al. (1995), novamente nota-se a vantagem do método *Tabu Search*. Nesse conjunto de dados, foi o único método que conseguiu diminuir inclusive o valor para uma das instâncias da literatura. Porém, na execução de algumas instâncias a heurística de *Clark e Wright* conseguiu ter melhor desempenho.

Os dados que mais chamam a atenção são os resultados da heurística de *Clark e Wright* e da *Tabu Search* para as instâncias A-n32-k5 e P-n101-k4. Na primeira, o resultado da execução para a *Tabu Search* tem um percentual de 0,4% com uma diferença grande quando relacionado aos 15,7% de *Clark e Wright*. Já no segundo, a vantagem é da heurística de *Clark e Wright*, que apresenta percentual de 10,6% contra 17,6% da *Tabu Search*.

Os dados de execução do algoritmo *Sweep* mantiveram-se acima dos demais métodos, com resultado muito inferior.

Quanto ao tempo, nota-se a mesma tendência das instâncias anteriores, com a *Tabu Search* apresentando valores acima dos demais. Além disso, em todos os casos, mesmo onde a meta-heurística *Tabu Search* teve desempenho pior que a de *Clark Wright*, o tempo de execução foi maior.

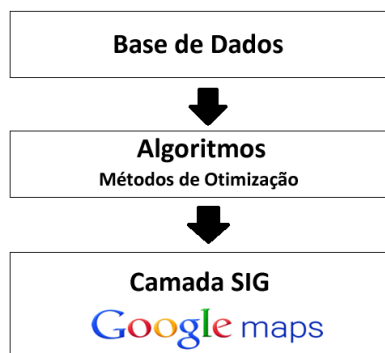
## 5.2 SISTEMA DE APOIO A DECISÃO PARA RESOLUÇÃO DE PRV COM INTERFACE SIG

Uma vez que os algoritmos propostos mostraram resultados satisfatórios, eles foram utilizados como componentes de otimização no sistema de apoio a decisão proposto nesse trabalho. Para o desenvolvimento do sistema, foram utilizadas tecnologias *web* que permitem que o mesmo seja executado em navegadores, não sendo necessária a sua instalação nos computadores que fazem seu uso. Entre as funcionalidades disponíveis no sistema de apoio a decisão estão a capacidade de gerar diferentes cenários de distribuição e visualização das rotas em ambiente georreferenciado.

### 5.2.1 Arquitetura do Sistema

Para atingir os objetivos propostos no desenvolvimento do sistema, o mesmo foi projetado levando em consideração três processos: entrada de dados, processamento e visualização das rotas. A entrada de dados é a etapa de carregamento dos dados e preparação do cenário para o processo de roteamento. A etapa de processamento consiste na seleção de algoritmos, variáveis de otimização e variáveis de restrição para a determinação das soluções e a terceira consiste na análise das rotas geradas utilizando a interface SIG. Essa estrutura é mostrada na Figura 24.

Figura 24 - Arquitetura do sistema



Fonte: O autor

O sistema computacional resultante desse trabalho foi desenvolvido utilizando a plataforma Java EE (*Java Enterprise Edition*) com a utilização da especificação *Java Server Faces* (JSF), para o desenvolvimento das interfaces e o *framework Hibernate* (HIBERNATE, 2014) para mapeamento do banco de dados relacional para o modelo orientado a objetos.

A plataforma JEE é baseada em padrões para o desenvolvimento de aplicações web, normalmente concebidas como aplicações multicamadas, com uma camada *frontend*, consistindo de um *framework web*, uma camada intermediária, a camada de controle, que proporciona segurança e controle de transações e uma camada *backend*, que fornece conectividade a um banco de dados ou sistema legado (GUPTA, 2013). Essas características foram importantes na seleção da linguagem e plataforma de desenvolvimento, pois são compatíveis com a arquitetura proposta para o sistema desenvolvido, que possui três camadas distintas: a camada de apresentação, a camada de negócios e a camada de persistência. A arquitetura do sistema está organizada conforme Figura 25.

Figura 25 - Arquitetura do sistema



Fonte: O autor

Na camada de apresentação, foi utilizada a especificação JSF para a definição dos componentes HTML (*Hypertext Markup Language*) que estruturam a interface. Para poder utilizar de componentes de interfaces mais sofisticados, foi utilizada a biblioteca de componentes *Primefaces* (PRIMEFACES, 2014). O *Primefaces* é uma biblioteca de componentes visuais que possui uma série de componentes com especificação CSS (*Cascading Style Sheets*) própria, além de fornecer a integração com a API do *Google Maps*, recurso necessário nesse sistema.

A camada de negócios é estruturada com as classes do sistema que fazem uso do pacote de algoritmos desenvolvidos nesse trabalho. As classes definidas nela implementam as regras de negócio necessárias para utilização dos algoritmos, fazendo a entrada de dados do sistema e fornecendo a saída para a visualização na camada SIG.

A camada de persistência de dados é utilizada para as operações de recuperação e gravação de dados em um banco de dados relacional. Para facilitar essa tarefa, é utilizado o *framework Hibernate* (HIBERNATE, 2014). Esse *framework*, segundo Gupta (2013), consiste em um conjunto de classes responsáveis pelas operações comuns de banco de dados utilizados em sistema d informação. Seu uso realiza abstrações necessárias para permitir independência de banco de dados e implementação de métodos abstratos de acesso aos dados.

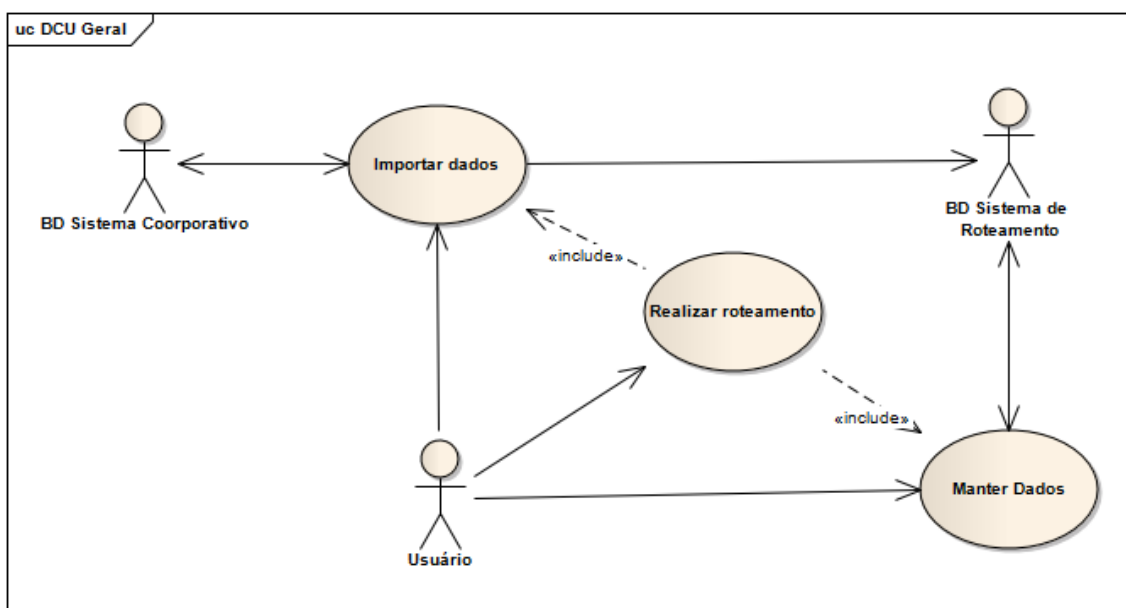
### **5.2.2 Modelagem do Sistema**

No processo de desenvolvimento foram seguidas as etapas de modelagem de software descritas na *Unified Modeling Language* (UML) (BOOCH; RUMBAUGH; JACOBSON, 1999). A UML foi utilizada para a modelagem dos artefatos da engenharias de software e para a modelagem de banco foi utilizando o modelo entidade relacionamento (MER). A UML é uma linguagem de modelagem visual adotada como padrão para modelagem orientada a objetos e design no desenvolvimento de software. É composta por um conjunto de diagramas que são usados para representar os modelos de um sistema e seus estágios de desenvolvimento (BEZERRA, 2006). O MER é apresentado para visualização do APENDICE I deste trabalho.

Para a modelagem do sistema foi utilizado o diagrama de casos de uso (*use case*). Segundo Bezerra (2006) o diagrama de casos de uso é uma representação das funcionalidades externamente observáveis do sistema e dos elementos que interagem com ele e faz parte da especificação de requisitos. Ele foi desenvolvido por apresentar uma visão do sistema, seus componentes e a interação entre eles. O diagrama de classes é utilizado para definir os componentes estruturais de um sistema orientado a objetos, definindo classes e a interação entre elas.

Foram desenvolvidos dois diagramas de casos de uso, um com visão geral do sistema, com a especificação de atores e suas principais ações e outro com nível maior de detalhamento do sistema e suas operações. Na Figura 26 é mostrado o diagrama geral de casos de uso, com a especificação dos atores e suas principais ações.

Figura 26 - Diagrama geral de casos de uso



Fonte: O autor

Os atores definidos durante a modelagem estão relacionados aos elementos internos e externos existentes no sistema. O carregamento inicial de dados para a implantação do sistema pode ser feito a partir de duas fontes: um banco de dados corporativo ou arquivos de texto com a listagem dos dados. Esses dados são processados e incorporados ao banco de dados próprio do

sistema de roteamento. Logo, na definição dos atores que interagem com o sistema, temos:

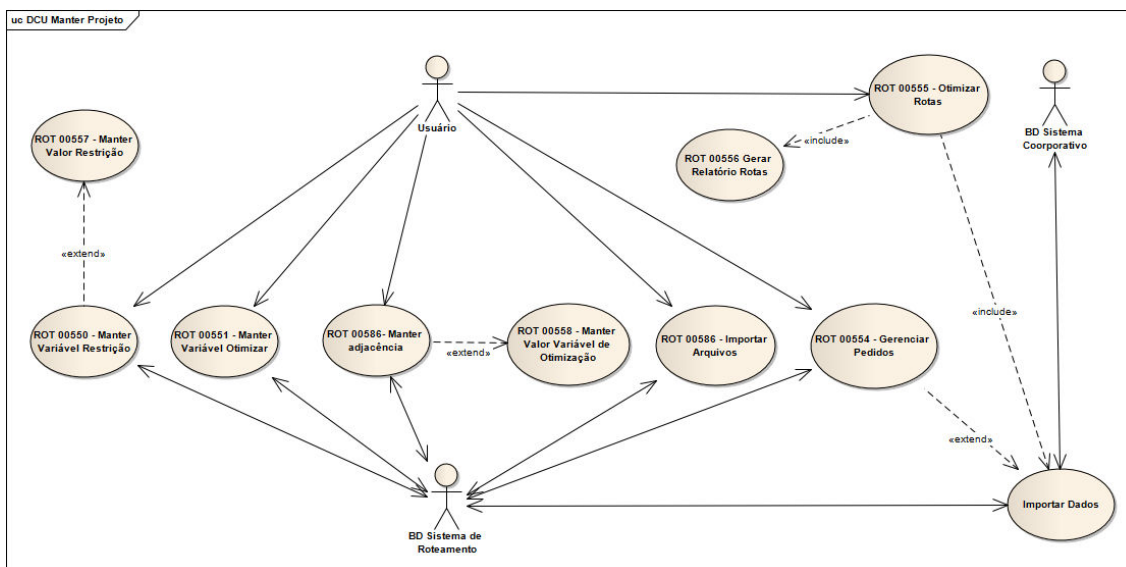
- Ator Usuário: É o agente externo que utiliza o sistema e realiza as operações de manutenção dos dados e geração de rotas;
- Ator BD Sistema Corporativo: É o agente interno que é acionado em caso de exportação de dados provenientes de banco de dados já existente na empresa que utiliza o sistema;
- Ator BD Sistema de Roteamento: É o agente interno do sistema, que representa o banco de dados. Além de ser o repositório de dados do sistema, também exerce a função de armazenar e executar algoritmos para o processamento de distâncias a partir de arquivos de texto vindos de GPS.

Nesse cenário, os casos de uso representam as ações que cada agente, interno ou externo, podem realizar no sistema. São eles:

- Importar Dados: São as operações de manipulação de dados existentes no sistema, seja a importação a partir de banco de dados ou arquivos de texto.
- Manter Dados: São as operações de manutenção dos dados das entidades que compõem o sistema.
- Realizar Roteamento: É o principal diagrama, que representa o processo de realização do roteamento, utilizando os outros diagramas como auxiliares para a importação dos dados que possibilitam a realização do roteamento.

No detalhamento do diagrama geral de casos de uso, conforme Figura 27, são mostradas as operações que envolvem o processo de operação do sistema.

Figura 27 - Diagrama de casos de uso



Fonte: O autor

O detalhamento dos casos de uso permite visualizar as operações do sistema disponíveis para cada usuário. O ator “Usuário” que representa o usuário do sistema de roteamento, pode realizar operações sobre as adjacências, variáveis de restrição e variáveis de otimização. As arestas são as junções entre propriedades que representam as arestas do grafo a ser otimizado e são relacionadas às variáveis de restrição e otimização. É possível gerenciar os dados sobre as variáveis otimização, que são aplicadas às arestas do sistema e compõem os objetos de minimização dos sistema e as restrições, que definem as regras que os algoritmos devem obedecer. Os atores “BD Sistema de Roteamento” e “BD Sistema Cooperativo” são responsáveis pela manutenção dos dados do sistema.

### 5.2.3 Visão Geral do Sistema

O resultado final deste trabalho foi o desenvolvimento de um sistema de apoio a decisão capaz de determinar rotas a partir da utilização dos algoritmos heurísticos e meta-heurísticos que possam ser utilizadas em ambientes reais de distribuição de produtos agrícolas, como fertilizantes, defensivos e produtos acabados. O sistema é composto por telas utilizadas na manutenção dos dados, pelo processo de otimização e pela camada SIG desenvolvida como ferramenta de visualização para auxiliar a tomada de decisão na distribuição de produtos.



Para ter uma visão geral do sistema e de seu funcionamento, as telas disponíveis no sistema estão resumidas no Quadro 4.

Quadro 4 - Relação de telas do sistema

| <b>Código</b> | <b>Nome</b>                   | <b>Descrição</b>                                                                                                                              |
|---------------|-------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------|
| ROT 00586     | Manter Adjacências            | Utilizado para manter as adjacências, que são as ligações entre os pontos que compõem o sistema                                               |
| ROT 00551     | Manter Variável de Otimização | Utilizado para manter os valores das variáveis de otimização do sistema                                                                       |
| ROT 00550     | Manter Variável de Restrição  | Utilizado para manter os valores das variáveis de restrição do sistema                                                                        |
| ROT 00556     | Manter Valores Restrição      | Utilizado para atualizar dados relacionados às variáveis de restrição                                                                         |
| ROT 00554     | Gerenciar Pedidos             | Utilizado para gerenciar os pedidos e controlar a demanda e o status de cada um                                                               |
| ROT 00586     | Importar Arquivos             | Utilizado para importar os arquivos de dados que permitem a atualização dos dados do sistema por meio de arquivos de texto                    |
| ROT 00555     | Otimizar Rotas                | Utilizado para realizar o processo de roteamento, permitindo selecionar algoritmos e restrições que compõem o processo de geração de soluções |

Fonte: O autor

O sistema consiste em um sistema *web* que permite gerenciar os dados de entrada e saída utilizados nos algoritmos de otimização. Ele pode utilizar dados provenientes de sistemas de informação já disponível e operando nos ambientes onde é aplicado o processo de otimização ou pode ser um sistema isolado, possuindo capacidade de cadastro de todos os dados necessários para a realização do processo de roteamento. Na Figura 28, é apresentada uma das interfaces do sistema, mostrando as opções de manutenção de dados disponíveis para cada caso de uso.

Figura 28 - Tela de manutenção de variáveis de adjacência

The screenshot shows the 'Sistema de Rotas' application interface. The main window is titled 'RO100586 - Manter Variável de Adjacências'. It features a search bar with 'Buscar' and 'Limpar' buttons. Below the search bar is a table with columns: 'Ações', 'Propriedade de Origem', 'Propriedade de Destino', 'Tipo de Via', 'Sentido Único', and 'Ativo'. The table contains two records:

| Ações  | Propriedade de Origem     | Propriedade de Destino | Tipo de Via     | Sentido Único | Ativo |
|--------|---------------------------|------------------------|-----------------|---------------|-------|
| [Icon] | 800 - TERRENO FOREIRO     | 5 - CHACARA SAO JOAO   | Entrada de Chão | Sim           | Sim   |
| [Icon] | 451 - CHACARA SANTA LUIZA | 475 - FAZENDA SAROJA   | Rua Asfaltada   | Não           | Não   |

Below the table, there is a 'Valor das variáveis de otimização' dialog box. It shows the selected adjacency and a table of optimization variables:

Adjacência Selecionada:

- Origem: TERRENO FOREIRO
- Destino: CHACARA SAO JOAO
- Via: Entrada de Chão
- Sentido Único: Sim
- Ativo: Sim

Table of optimization variables:

| Variável        | Ativo | Valor Mínimo | Valor Máximo | Valor  |
|-----------------|-------|--------------|--------------|--------|
| Teste 270114001 | Sim   | 100,00       | 200,00       | 100,00 |
| Teste 270114002 | Sim   | 1,00         | 5,00         | 1,00   |
| Teste 270114003 | Sim   | 5,00         | 25,00        | 5,00   |

Fonte: O autor

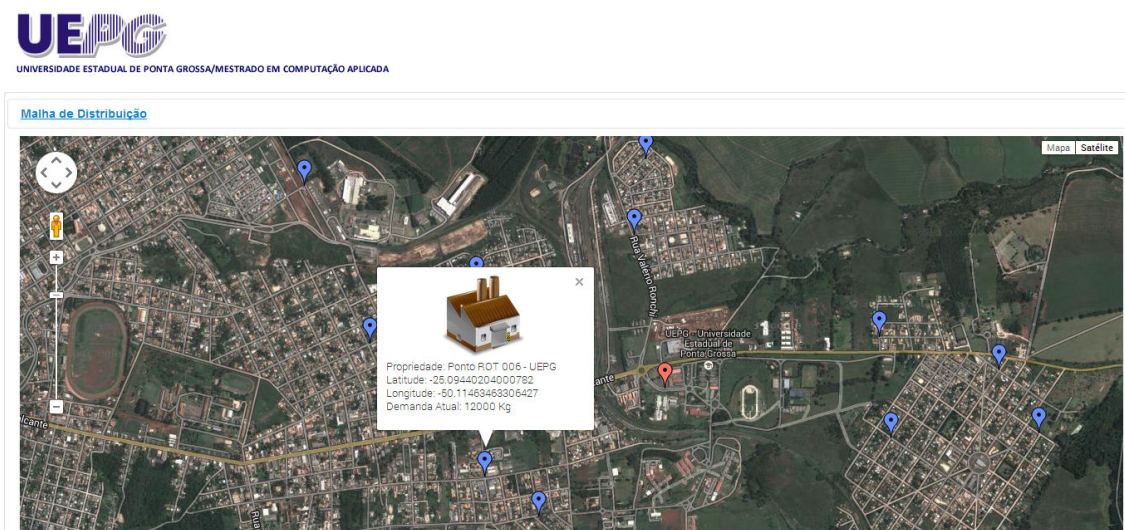
A tela mostrada na Figura 28 representa o padrão utilizado em todo o sistema, com componentes estruturados usando *Primefaces* de acordo com as operações de cada tabela. Esse padrão de tela é utilizado em todo o sistema.

#### 5.2.4 Camada de Visualização SIG

Para cumprir o objetivo de fornecer uma interface visual que facilite a visualização dos dados de pontos e das rotas geradas pelos algoritmos foi desenvolvida uma interface SIG baseada na API do *Google Maps*<sup>®</sup>. Assim como em Santos (2011) e Casal (2012) essa camada de visualização proporciona um nível de representação cartográfica de relevância em sistemas para resolução de problemas de roteamento aplicados em situações reais.

A camada de visualização possui os recursos nativos, já presentes na própria API do *Google Maps*<sup>®</sup>, como a possibilidade de aumentar e diminuir o grau de aproximação do mapa, alternar entre os modos de visualização de satélite, conforme Figura 29 ou o modo de visualização de mapas como na Figura 30. Os demais recursos foram definidos de acordo com as necessidades do problema de roteamento de veículos.

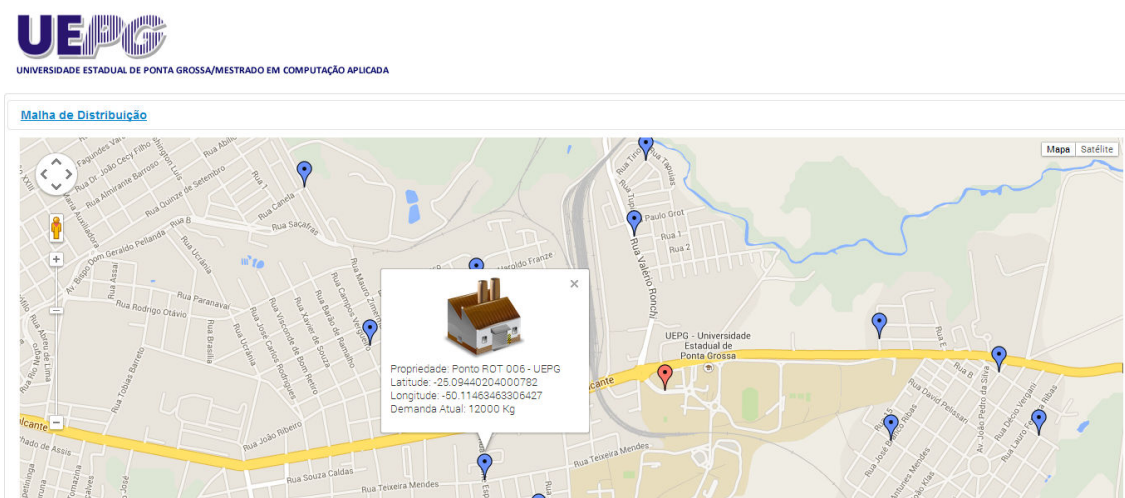
Figura 29 - Visualização na camada SIG do sistema sem roteamento das entregas.



Fonte: O autor.

Na Figura 29 é mostrado um exemplo de visualização da malha antes da execução do roteamento. Os pontos em azul representam cada uma das propriedades e o único ponto em vermelho, representa o ponto de partida para distribuição dos produtos entregues, definido no PRV como ponto de origem. Clicando sobre os marcadores disponíveis no mapa, que representam cada ponto, é possível ver as informações sobre eles, como nome, latitude, longitude e a demanda de entrega para o dia atual.

Figura 30 - Visualização da camada SIG usando o modo de mapas.

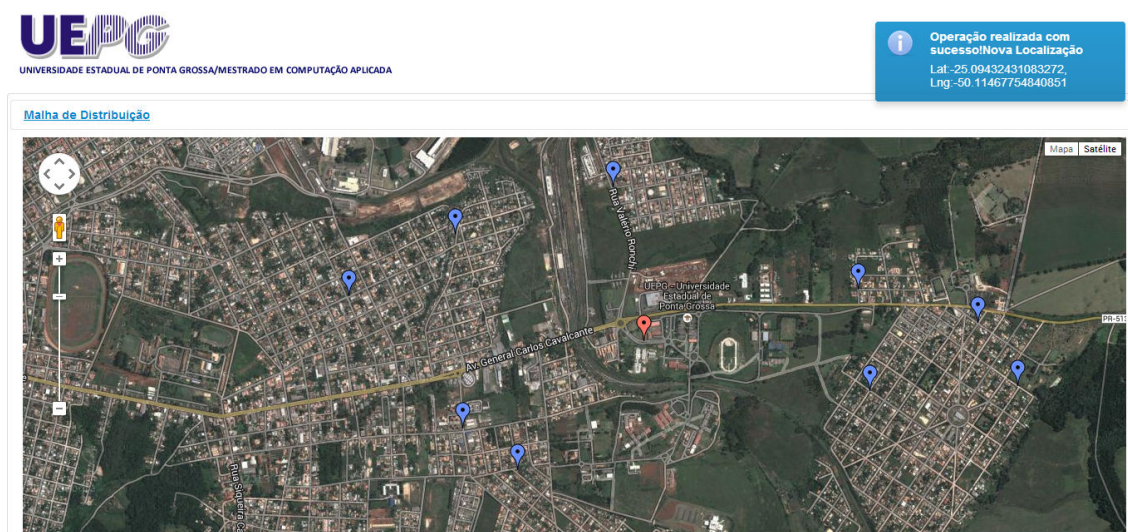


Fonte: O autor.





Figura 32 - Resultado da operação de atualização de coordenadas utilizando a interface SIG.



Fonte: O autor.

Como apresentado, a interface SIG desenvolvida serve não somente para a visualização de dados e de rotas, mas também para operações de atualização e mapeamento de pontos de entrega.

## 6 DISCUSSÃO

Os resultados dos algoritmos desenvolvidos nesse trabalho, assim como em Gendreau (1999), Brandão (2009) e Montané (2006), mostram que as meta-heurística *Tabu Search* são melhores quando comparados com métodos heurísticos. No caso estudado nesse trabalho, onde foram utilizadas as heurísticas de *Clark e Wright* e *Sweep* essa tendência se manteve. Pelas características apresentadas por Glover e Laguna (1997), esse resultado era esperado, uma vez que as meta-heurísticas exploram de maneira mais ampla o espaço de soluções, evitando cair em ótimos locais. Porém, os resultados apresentados pela *Tabu Search* ainda podem ser melhorados, pois os resultados em relação aos valores ótimos relatados na literatura apresentam melhores resultados na maioria dos casos. Assim como em Repoussis (2010), que aplicando um método meta-heurístico nas instâncias F-n45-k4 e F-n135-k7 de Fisher conseguiu resultados melhores que os apresentados pela *Tabu Search* neste trabalho, indiciando que o estudo e aplicação de meta-heurísticas é promissor na busca por melhores resultados.

Quanto a heurística *Sweep* os resultados foram inferiores em relação aos outros métodos. Geralmente esse método é utilizado como geração de soluções iniciais para métodos heurísticos, como em Tang (2010). O método também foi utilizado por Gandelman (2010), onde é apresentado como um método inicial de geração de soluções para PRV, onde os resultados apresentam desempenho similar a esse trabalho, sendo melhorados somente após uma etapa de melhoria, não presente nesse trabalho. Apesar dos resultados com desempenho pouco satisfatório, o método *Sweep* foi utilizado nesse trabalho pelas características das rotas geradas baseado no processo de clusterização e posterior otimização. Do ponto de vista de distribuição em ambientes reais, o agrupamento realizado por este algoritmo pode ser considerado relevante e possível de ser utilizado, apesar dos resultados apresentados nesse trabalho.

O sistema de apoio a decisão desenvolvido neste trabalho, assim como os propostos por Casal (2012), Santos (2011) e Ruiz (2004), tem em comum o desenvolvimento e validação de métodos de otimização combinatória para a aplicação nos sistemas. No sistema desenvolvido por Ruiz (2004), existe a

ausência da camada SIG, com rotas e resultados de percursos mostrados somente em listagens de dados. A ausência da camada não interfere na determinação das rotas, porém, ao adicionar esse componente ao sistema, a tarefa de visualização de rotas é facilitada, recurso relevante em problemas reais de entrega de produtos.

Em relação a camada SIG desenvolvida, quando comparada a implementações de Casal (2012) e Santos (2011) a principal diferença está na maneira como são calculadas as distâncias e são formadas as rotas. No sistema desenvolvido nesse trabalho, onde o objetivo é a aplicação em ambientes rurais, a API do Google Maps não possui registro de todas as estradas, portanto, não consegue, com seus próprios recursos, realizar o cálculo de distâncias e rotas entre os pontos. Para isso, o sistema conta com algoritmos de cálculo das distâncias entre os pontos. Esse cálculo não é preciso, mas permite que sejam computadas as distâncias entre os pontos, traçando uma linha reta entre os mesmos. Para amenizar esse problema, o sistema aceita como entrada de dados, arquivos vindos de aparelhos GPS que fazem o registro preciso das estradas que são percorridas e, por meio de algoritmos específicos, permite o cálculo das distâncias entre os pontos de maneira mais precisa, considerando a real ligação entre os pontos.

## 7 CONCLUSÕES

Conclui-se com o trabalho que os algoritmos desenvolvidos para otimização atendem o objetivo de organizar o processo de entrega de produtos a partir de métodos de resolução de problemas de roteamento de veículos, de acordo com os resultados dos testes realizados. Assim, como relatado em outros estudos, o método meta-heurístico apresentou resultados melhores quando comparado aos métodos heurísticos. Porém, como o objetivo é verificar a viabilidade das rotas em ambiente real, a aplicação de todos os métodos no sistema de apoio a decisão desenvolvido nesse trabalho permite avaliar aspectos da malha viária para selecionar a solução que melhor se adapte ao ambiente real de distribuição.

O objetivo principal do trabalho, o desenvolvimento de um sistema de apoio a decisão para aplicação em problemas reais de roteamento de veículos, resultou em um sistema funcional e extensível a diferentes cenários de distribuição, permitindo que o mesmo seja aplicado a ambientes com restrições de caminhões e frotas homogêneas.

Considerando os objetivos secundários do trabalho, pode-se concluir que a camada SIG desenvolvida cumpre o objetivo de fornecer ao sistema uma interface de visualização que facilita não somente como componente visual, mas também na manutenção dos dados referentes aos pontos atendidos pelos algoritmos.



## REFERÊNCIAS

- ARCHETTI, C., SPERANZA, M. G. The Split Delivery Routing Problem: A Survey. **Operations Research/Computer Science Interfaces Series**. v. 43, p. 103-122, 2008.
- AUGERAT, P., et al. **Computational results with a branch and cut code for the capacitated vehicle routing problem**. (Technical Report RR 949-M). Universite Joseph Fourier, Grenoble, France, 1995.
- BALLOU, R. H. **Gerenciamento da Cadeia de Suprimentos: Planejamento, Organização e Logística Empresarial**. São Paulo: Bookman, 2001. 616 p.
- BELFIORE, P.P. **Scatter search para Problemas de Roteirização de Veículos com Frota Heterogênea, Janelas de Tempo e Entregas Fracionadas**. 2006, 203 f. Tese (Doutorado em Engenharia de Produção) - Escola Politécnica da Universidade de São Paulo, São Paulo, 2006.
- BEZERRA, E. **Princípios de Análise e Projeto de Sistemas com UML: Um guia prático para modelagem de sistemas**. Rio de Janeiro: Campus, 2006. 380 p.
- BOOCH, G.; RUMBAUGH, J.; JACOBSON, I. **UML: guia do usuário**; tradução de Fábio Freitas da Silva. Rio de Janeiro: Campus, 2000. 499 p.
- BODIN, L. D. Twenty Years of Routing and Scheduling. **Operations Research**. v. 30, p. 571-579, 1990.
- BRANDÃO, J. A Deterministic Tabu Search Algorithm For The Fleet SIZE And Mix Vehicle Routing Problem. **European Journal of Operational Research**. v. 195, p. 716-728, 2009.
- CACCETTA, L., ALAMEEN, M., ABDUL-NIBY, M. An Improved Clarke and Wright Algorithm to Solve the Capacitated Vehicle Routing Problem. **Engineering, Technology & Applied Science Research**. v. 3, n. 2, p. 413-415, 2013.
- CAMPELO JÚNIOR, J. U. **Proposta de Otimização da Roteirização Distritos dos Carteiros: Um Estudo de Caso no Centro de Entregas Encomendas de Fortaleza**. 2010, 97 f. Dissertação (Programa e Mestrado em Logística e Pesquisa Operacional) - Universidade Federal do Ceará, Fortaleza, 2010.
- CARIĆ, T. et al. A Modelling and Optimization Framework for Real-World Vehicle Routing Problems. In: \_\_\_\_\_. **Vehicle Routing Problem**. Vienna: In-Tech, 2008. 15-34.

CASAL, J. A. V. **Vehicle Routing Problem - Investigação e construção de um Sistema de Informação Geográfica**. 2012, 101 f. Dissertação (Mestrado em Engenharia de Sistemas) - Universidade do Minho, Braga, 2012.

CHAVES, A. C. **Uma Meta-Heurística Híbrida Com Busca Por Agrupamentos Aplicada A Problemas De Otimização Combinatória**. 2009, 199 f. Tese (Doutorado em Computação Aplicada) - Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 2009.

CHRISTOFIDES, N., EILON, S. An algorithm for the vehicle dispatching problem. **Operational Research Quarterly**, v. 20, p. 309-318, 1969.

CLARKE, G.; WRIGHT J.W. Scheduling Of Vehicles From A Central Depot To A Number Of Delivery Points. **Operations Research**, v. 12, p. 568-581, 1962.

COIN. Disponível em <<http://www.branchandcut.org/>> Acesso em: 03 fev. 2014.

CORDEAU, J., LAPORTE, G., MERCIER, A. A Unified Tabu Search Heuristic For Vehicle Routing Problems With Time Windows. **The Journal of the Operational Research Society**. v. 52, p. 141-145, 2001.

CORDEAU, J. et al. A Guide to Vehicle Routing Problem. **The Journal of the Operational Research Society**, v. 53, n. 5, p. 512-522, 2002.

CORDEAU, J. et al. New Heuristics for the Vehicle Routing Problem. In: LANGEVIN, A e RIPOEL, D. **Logistics Systems: Design and optimization**. OnLine: Springer, 2005. p. 279-297.

CUNHA, C. B. Aspectos Práticos da Aplicação de Modelos de Roteirização de Veículos a Problemas Reais. **Transportes**, V.8, n.2, p. 51-74, 2003.

DANTZIG, G. B.; RAMSER, J.H. The Truck Dispatching Problem. **Management Science**, v. 6, p. 80-91, 1959.

DROR, M., LAPORTE, G., TRUDEAU, P. Vehicle routing with split deliveries. **Discrete Applied Mathematics**, v. 50, n. 3, p. 239-254, 1994.

ECLIPSE. Disponível em <<http://www.eclipse.org/>> Acesso em: 26 jan. 2014.

FISHER, M. L. Optimal Solution of Vehicle Routing Problems Using Minimum K-trees, **Operations Research**, v. 42, p. 626-642, 1994.

FISHER, M. L.; JAIKUMAR, R. A Generalized Assignment Heuristic for Vehicle Routing. **Networks**, v. 11, p. 109-124, 1981.

GALDELMAN, D. A. **Busca Dispersa Aplicada ao Problema de Roteamento de Veículos com Múltiplos Depósitos**. 2010, 69 f. Dissertação (Programa de Pós-Graduação em Engenharia de Produção) - Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2010.

GENDREAU, M., HERTZ, A., LAPORTE, G. A Tabu Search Heuristic for the Vehicle Routing Problem. **Management Science**, v. 40, n. 10, p. 1276-1290, 1994.

GENDREAU, M. et al. A Tabu Search Heuristic For The Heterogeneous Fleet Vehicle Routing Problem. **Computers and Operations Research**, v. 26, p. 1153-1173, 1999.

GLOVER, F. Future Paths For Integer Programming and Links To Artificial Intelligence. **Computers and Operations Research**. v. 5, p. 553-549, 1986.

\_\_\_\_\_. Tabu Search - Part I. **ORSA Journal on Computing**. v. 1, p. 190-206, 1989.

\_\_\_\_\_. Tabu Search - Part II. **ORSA Journal on Computing**. v. 2, n. 1, p. 190-206, 1990.

GLOVER, F., LAGUNA, M. **Tabu Search**. Boston : Kluwer Academic Publishers, 1997. 408 p.

GOLDBARG, M.C. **Otimização combinatória e programação linear: modelos e algoritmos**. Rio de Janeiro: Elsevier, 2000. 649 p.

GOOGLE MAPS. Disponível em <<https://developers.google.com/maps/?hl=pt-br>>. Acesso em: 05 fev. 2014.

GOMES, R. F. S. **Aplicação da Metaheurística Tabu Search Na Otimização de Rotas e Manutenção Preventiva em Campo**. 2011, 108 f. Dissertação (Programa de Mestrado em Logística e Pesquisa Operacional) - Universidade Federal do Ceará, Fortaleza, 2011.

GUERRERO, A. F. M. **Construção de uma Metaheurística de Otimização de Rotas de Veículos**. 2009, 86 f. Dissertação (Mestrado em Engenharia e Gestão Industrial) - Universidade Técnica de Lisboa, 2009.

GUPTA, A. **Java EE7 Essenciais**. Califórnia: O'Reilly, 2013. 362 p.

HIBERNATE. Disponível em <<http://hibernate.org/>>. Acesso em: 14 fev. 2014.

JAVA. Disponível em <<http://www.oracle.com/technetwork/java/index.htm>>. Acesso em: 26 jan. 2014.

LAGUNA, M. A Guide to Implementing Tabu Search. **Investigación Operativa**. v. 4, n. 1, p. 5-25, 1994.

LAPORTE, G. The Vehicle Routing Problem: An overview of exact and approximate algorithms. **European Journal of Operational Research**. v. 59, p. 345-358, 1992.

\_\_\_\_\_. Fifty Years of Vehicle Routing. **Transportation Science**. v. 43, n. 4, p. 408-416, 2009.

LAPORTE, G., SEMET, F. Classical Heuristics for the Capacitated VRP. In: TOTH, P & VIGO, D. **The Vehicle Routing Problem**. Philadelphia: Society for Industrial and Applied Mathematics, 2001. p. 109-128.

LIAO, T., HU, T. An Object-Oriented Evaluation Framework For Dynamic Vehicle Routing Problems Under Real-Time Information. **Expert Systems with Applications**. v. 38, n. 10, p. 12548-12558, 2011.

LIMA, F. Q. **Algoritmos Para problemas Reais de Roteamento de Veículos: Uma Análise Comparativa**. 2005. 111 f. Dissertação (Programa de Pós-Graduação em Ciência da Computação) - Universidade Federal Fluminense, Rio de Janeiro, 2005.

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

MONTANÉ, F. A. T., GALVÃO, R. D. A Tabu Search Algorithm For The Vehicle Routing Problem With Simultaneous Pick-Up And Delivery Service. **Computers e Operations Research**. v. 33, p. 595-619, 2006.

NASCIMENTO, I. Z. **Abordagens Determinísticas e Estocásticas Para o Problema de Roteirização de Veículos na Entrega de Refeições**. 2011, 99 f. Dissertação (Programa de Pós-Graduação em Métodos Numéricos e Engenharia) - Universidade Federal do Paraná, Curitiba, 2011.

NEO. Disponível em <<http://neo.lcc.uma.es/>>. Acesso em: 03 fev. 2014.

OSMAN, I.H. Metastrategy simulated annealing and tabu search algorithms for the vehicle routing problem. **Annals of Operations Research**, v. 41, n. 4, p. 421-451, 1993.

OSMAN, I. H., LAPORTE, G. Metaheuristics: A bibliography. **Annals of Operations Research**. v. 63, n. 5, p. 511-623, 1996.

PRIMEFACES. Disponível em < <http://primefaces.org/> >. Acesso em: 16 fev. 2014.

QUEIROZ, M. A. V. **Problema de Roteirização de Veículos com Janelas de Atendimento, Frotas Heterogêneas e Entregas Fracionadas**. 2012, 113 f. Dissertação (Mestrado em Ciência da Computação) - Universidade Estadual de Londrina, Centro de Ciências Exatas, Londrina, 2012.

REINELT, G. **TSPLIB**. Disponível em <<http://comopt.ifl.uni-heidelberg.de/software/TSPLIB95/>>. Acesso em: 04 fev. 2014.

RENAUD, J., BOCTOR, F. F., LAPORTE, G. An improved petal heuristic for the vehicle routing problem. **Journal of the Operational Research Society**. v. 47, p. 329-336, 1996.

REPOUSSIS, P. P. et al. A Hybrid Evolution Strategy For The Open Vehicle Routing Problem. **Computers e Operations Reearch**. v. 37, p. 443-455, 2010.

RUIZ, R., MAROTO. C., ALCARAZ, J. A Decision Support System For a Real Vehicle Routing Problem. **European Journal of Operational Research**. v. 153, p. 593-606, 2004.

SANTOS, L., COUTINHO-RODRIGUES, J., ANTUNES, C.H. A web spatial decision support system for vehicle routing using Google Maps. **Decision Support Systems**, v. 51, n.1, p. 1-9, 2011.

SOSA, N. G. M., GALVÃO, R. D., GANDELMAN, D. A. Algoritmo de Busca Dispersa Aplicado ao Problema Clássico de Roteamento de Veículos. **Pesquisa Operacional**, v. 27, n. 2, p. 293-310, 2007.

SOLOMON. Disponível em <<http://w.cba.neu.edu/~msolomon/home.htm>>. Acesso em: 03 fev. 2014.

SOUZA, M. J. F. **Inteligência Computacional para Otimização**. Departamento de Computação, Universidade Federal de Ouro Preto, 2011.

TANG, J., Zhang, J., PAN, Z. A Scatter Search Algorithm For Solving Vehicle Routing Problem With Loading Cost. **Expert Systems with Applications**. v. 37, p. 4073-4083, 2010.

TIBURCIO, D. M. **Técnicas da Pesquisa Operacional na Abordagem do Problema de Roteamento no Transporte de Funcionários de Empresas**. 2012, 86 f. Dissertação (Programa de Pós-Graduação em Engenharia de Produção) - Universidade Estadual do Paraná, Curitiba, 2012.

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

VIANA, F. H. F. **Algoritmo para o Problema de Roteamento Dinâmico de Veículos com Janelas de Tempo e Tempos de Viagem Variável**. 2007, 88 f. Dissertação (Programa de Pós-Graduação em Ciência da Computação) - Universidade Federal de Minas Gerias, Belo Horizonte, 2007.