

UNIVERSIDADE REGIONAL DE BLUMENAU
CENTRO DE CIÊNCIAS EXATAS E NATURAIS
CURSO DE CIÊNCIA DA COMPUTAÇÃO – BACHARELADO

**ROTEIRIZAÇÃO DE VEÍCULOS PARA REALIZAÇÃO DE
COLETA UTILIZANDO ALGORITMOS EVOLUCIONÁRIOS**

CLEBER JOSÉ OSIKA

BLUMENAU
2014

2014/1-03

CLEBER JOSÉ OSIKA

**ROTEIRIZAÇÃO DE VEÍCULOS PARA REALIZAÇÃO DE
COLETA UTILIZANDO ALGORITMOS EVOLUCIONÁRIOS**

Trabalho de Conclusão de Curso submetido à
Universidade Regional de Blumenau para a
obtenção dos créditos na disciplina Trabalho
de Conclusão de Curso II do curso de Ciência
da Computação — Bacharelado.

Prof. Cláudio Ratke, Mestre - Orientador

**BLUMENAU
2014**

2014/1-03

ROTEIRIZAÇÃO DE VEÍCULOS PARA REALIZAÇÃO DE COLETA UTILIZANDO ALGORITMOS EVOLUCIONÁRIOS

Por

CLEBER JOSÉ OSIKA

Trabalho aprovado para obtenção dos créditos
na disciplina de Trabalho de Conclusão de
Curso II, pela banca examinadora formada
por:

Presidente: _____
Prof. Cláudio Ratke Orientador, Mestre – Orientador, FURB

Membro: _____
Prof. Mauro Marcelo Mattos, Doutor – FURB

Membro: _____
Prof. Antônio Carlos Tavares, Mestre – FURB

Blumenau, 07 de julho de 2014.

Dedico este trabalho a minha família, amigos e especialmente aqueles que me ajudaram diretamente na realização deste.

AGRADECIMENTOS

À minha família, cujo meu bem mais precioso.

Aos meus amigos, pela compreensão de minha ausência em função do tempo dedicado ao trabalho e por sempre me apoiarem em momentos difíceis.

Ao meu orientador, Cláudio Ratke, me direcionando na resolução do trabalho e acreditando na conclusão do mesmo.

Se você quer ser bem sucedido, precisa ter dedicação total, buscar o seu último limite e dar o melhor de si.

Ayrton Senna

RESUMO

Este trabalho apresenta o desenvolvimento de um sistema web de logística para coletar mercadorias, que tem por objetivo minimizar o custo de transporte encontrando as melhores rotas e representando-as graficamente em um mapa. O problema de roteirização estudado neste trabalho utiliza restrições de capacidade de veículo, janela de tempo e restrições dinâmicas baseada no modelo de domínio do sistema. O método de resolução proposto foi a heurística algoritmo genético, que é muito utilizada em problemas complexos de explosão combinatória. Os resultados obtidos demonstram que o sistema é eficiente para encontrar uma boa rota e o modelo é flexível referente as restrições impostas no problema e futuras.

Palavras-chave: Problemas de roteamento de veículos. Algoritmo genético. Logística.

ABSTRACT

This paper presents the development of web logistics system to collect products, whose objective to minimize the transport cost and finding the best routes and represent them graphically on a map. The routing problem studied in this paper uses vehicle capacity constraints, time windows and dynamic constraints based on the domain model of the system. The method proposed of resolution was genetic algorithm, which is widely used in complex problems of combinatorial explosion. The results show that the system is efficient to find good route and the model is flexible the restrictions imposed the problem and future.

Key-words: Vehicle routing problems. Genetic algorithm. Logistics.

LISTA DE ILUSTRAÇÃO

Figura 1 – Cadeia de abastecimento	18
Figura 2 – Três nós percorridos por 2 veículos	20
Figura 3 – Três nós percorridos por um veículo.....	21
Figura 4 – Mapa da Austrália	22
Quadro 1 – Definição e Solução do problema de coloração de mapas modelado em CSP.....	22
Figura 5 – Fluxo de controle do algoritmo genético	24
Figura 6 – Exemplo de população de indivíduos	25
Figura 7 – Representação dos pais com o ponto de corte definido	26
Figura 8 – Cópia das seções mapeamento dos pais para os filhos.....	26
Figura 9 – Cópia dos genes restantes dos pais para os filhos.....	27
Figura 10 – Mutação simples.....	27
Figura 11 – Diagramas de caso de uso	32
Figura 12 – Diagrama de classes da camada de domínio do sistema.....	34
Figura 13 – Diagrama de classes da camada de <i>engine</i>	35
Figura 14 - Diagrama de atividades do sistema.....	36
Figura 15 – Diagrama de atividade da rotina roteirizar.....	37
Quadro 2 - Fórmula	38
Quadro 3 – calcular aptidão do cromossomo	40
Quadro 4 – Início da Rotina de cruzamento	41
Quadro 5 – Continuação da Rotina de cruzamento	42
Quadro 6 – rotina de mutação.....	42
Quadro 7 – Método que calcula o custo em km e Tempo entre dois pontos.....	43
Quadro 8 – Rotina de roteirização consumida da <i>WebService</i> da API do Bing.....	44
Quadro 9 – Rotina que recebe os pontos geográficos	45
Quadro 10 – Rotina que gera o mapa	45
Quadro 11 – nova geração de cromossomos	46
Figura 16 - Menu do sistema desenvolvido.....	46
Figura 17 - Tela de depósito.....	47
Figura 18 – Menu principal da ferramenta	47
Figura 19 – Menu cadastros e subitens.....	47
Figura 20 – Menu cadastros de endereços e subitens.....	48

Figura 21 – Menu configurações roteirizador	48
Figura 22 – Cadastro de cliente	49
Figura 23 – Cadastro de produtos.....	50
Figura 24 – Cadastro de veículos	51
Figura 25 – Configurações do algoritmo genético	52
Figura 26 – Parametrizar restrição fixas.....	52
Figura 27 – Lista de restrição dinâmicas	53
Quadro 12 – Restrições dinâmicas cadastradas	53
Figura 28 – Formulário de Restrição dinâmica	54
Quadro 13 – Listagem de atributos disponíveis para montar restrições.....	55
Figura 29 – Formulário de Pedido de Coleta.....	56
Figura 30 - Itens do Pedido	57
Figura 31 - Roteirização	57
Figura 32 – Resultado da roteirização	58
Figura 33 - Mapa da rota número um gerada	59
Figura 34 – Mapa da rota número dois gerada	59
Quadro 14 - Comparação com trabalhos correlatos	60
Quadro 15 – Comparação dos resultados obtidos com 15 pontos de coletas	61
Quadro 16 – Parâmetros obtidos dos resultados com 15 coletas.....	61
Quadro 17 - Comparação dos resultados obtidos com 24 pontos de coletas.....	61
Quadro 18 – Parâmetros do roteiro com 24 pontos de coletas	62
Quadro 19 – Testes na execução do AG.....	63
Figura 35 Clientes do estudo de caso 1	69
Figura 36 – Matriz de distância do estudo de caso 1.....	70
Figura 37 - Clientes do estudo de caso 2	71
Figura 38 - Matriz de distância do estudo de caso 2	72

LISTA DE SIGLAS

API – *Application Programming Interface*

CEP - Código de Endereçamento Postal

CSP – Constraint Satisfaction Problem

EA – *Enterprise Architect*

HTTP – *Hypertext Text Transfer Protocol*

JSON – *JavaScript Object Notation*

MVC – *Model View Controller*

PMX – Cruzamento de Mapeamento Parcial

PRV – Problema de Roteirização de Veículo

km – Quilômetro

RF – Requisito Funcional

RNF – Requisito Não Funcional

SQL - *Structured Query Language*

UML – *Unified Modeling Language*

VPR - *Vehicle Routing Problem*

WCF - *Windows Presentation Foundation*

WEB - *World Wide Web*

SUMÁRIO

1 INTRODUÇÃO.....	14
1.1 OBJETIVOS DO TRABALHO	15
1.2 ESTRUTURA DO TRABALHO	15
2 FUNDAMENTAÇÃO TEÓRICA	17
2.1 LOGÍSTICA	17
2.1.1 Gestão de Transporte.....	19
2.2 PROBLEMA DE ROTEIRIZAÇÃO DE VEÍCULO (PRV)	19
2.3 CLARKE E WRIGHT.....	20
2.4 CONSTRAINT SATISFACTION PROBLEM (CSP).....	21
2.5 ALGORITMOS GENÉTICOS	22
2.5.1 Representação Cromossomial	24
2.5.2 População	24
2.5.3 Elitismo	25
2.5.4 Seleção de Cromossomo	25
2.5.5 Cruzamento	25
2.5.5.1 Cruzamento de mapeamento parcial PMX	26
2.5.6 Mutação.....	27
2.6 BING MAPS	28
2.7 TRABALHOS CORRELATOS	28
2.7.1 Roteirização de veículo	28
2.7.2 Sistema para logística de distribuição	29
2.7.3 Modelo de roteirização de veículos.....	30
3 DESENVOLVIMENTO	31
3.1 REQUISITOS PRINCIPAIS DO PROBLEMA A SER TRABALHADO.....	31
3.2 ESPECIFICAÇÃO	31
3.2.1 Diagrama de Casos de Uso	32
3.2.2 Diagrama de classes	33
3.2.2.1 Diagrama de classes simplificado da camada de Domínio.....	33
3.2.2.2 Diagramas de classe da camada <i>Engine</i> do sistema	35
3.2.3 Diagrama de atividades	36
3.2.3.1 Gerar população inicial.....	37

3.2.3.2 Ordenar população	37
3.2.3.3 Calcular aptidão	38
3.2.3.4 Nova geração	39
3.2.3.5 Separar Elitização	39
3.2.3.6 Operadores genéticos	39
3.2.3.7 Critério de parada	39
3.3 IMPLEMENTAÇÃO	40
3.3.1 Técnicas e ferramentas utilizadas.....	40
3.3.2 Desenvolvimento do sistema	40
3.3.2.1 Calcular aptidão	40
3.3.2.2 Cruzamento.....	40
3.3.2.3 Mutação	42
3.3.2.4 Calcular distância e tempo entre uma origem e um destino	42
3.3.2.5 Rotina de roteirização do Bing	43
3.3.2.6 Rotina de geração de um mapa no Bing Maps	44
3.3.2.7 Chamada da rotina nova geração	45
3.3.3 Operacionalidade da implementação	46
3.3.3.1 Cadastros.....	47
3.3.3.2 Roteirizar Pedidos.....	57
3.4 RESULTADOS E DISCUSSÃO	59
4 CONCLUSÕES.....	64
4.1 EXTENSÕES	64
REFERÊNCIAS	66
ANEXO A – Informações dos clientes do estudo de caso 1	69
ANEXO B – Matriz de distância do estudo de caso 1	70
ANEXO C – Informações dos clientes do estudo de caso 2	71
ANEXO D – Matriz de distância do estudo de caso 2	72

1 INTRODUÇÃO

Na maioria das empresas e indústrias o transporte é uma das atividades mais importante na composição do custo logístico, onde chega a custar até 60% dos gastos totais (RODRIGUES, 2007, p. 16). Portanto, uma das preocupações das empresas é de minimizar o custo do escoamento dos seus produtos com eficiência e qualidade.

Logística é uma área da gestão responsável pelos recursos, matérias-primas, componentes, equipamentos, serviços e informações necessárias para a execução e controle das atividades das organizações. É necessário coordenar e planejar a movimentação de todos esses itens, na busca pela melhor condição de operação no menor tempo possível (DIAS, 2010, p. 1).

O aumento de concorrência entre as empresas transformou a logística em uma importante fonte de vantagem competitiva. A sua importância nos processos da cadeia de suprimento está relacionada com agregação de valores e disponibilidade dos seus bens e serviços para os clientes, fornecedores da empresa e os demais interessados. Ou seja, um serviço ou produto apenas tem valor quando ele está disponível para ser consumido independente do lugar que o interessado esteja (TSUDA, 2007, p. 15).

As decisões relacionadas ao transporte são fundamentais no planejamento estratégico de logística e o seu principal problema está relacionado à roteirização de veículos (TSUDA, 2007, p. 15), onde normalmente, o transporte representa de um a dois terços dos custos logísticos totais (BALLOU, 2001, p. 149).

Roteamento de veículos é um problema que visa minimizar custos de transporte, encontrando a rota menos custosa para atender uma série de clientes e demais imposições que fazem parte do problema. Segundo Miura (2003, p. 10) a utilização de sistema de roteirização para problemas reais, pode gerar grandes benefícios na redução de custos, tempo de transporte, produtividade operacional, custos com agregados e uma melhor satisfação do cliente, já que o custo do escoamento de qualquer produto tem influência direta no preço final dele (ASSIS, 2007, p. 15).

Para as transportadoras a resolução de problemas de roteamento de veículos torna-se muito importante, pois essa é a sua atividade central. Segundo a Confederação Nacional de Transportes (2006 apud ASSIS, 2007, p. 15), apenas 5% das grandes empresas de transporte rodoviários utilizam sistema de informação como roteirizadores para minimizar os seus custos.

Para encontrar a rota menos custosa, é necessário calcular as possíveis combinações de um determinado problema. Contudo, dependendo do número de combinações isto pode requerer um processamento elevado, demorando muito para encontrar a solução ótima. Esse tipo de problema se encaixa na categoria de *NP-Hard* (CUNHA, 2000, p. 3). Nesta classificação de problema não existe uma forma matematicamente conhecida de ser resolvido.

Neste trabalho, há necessidade de construir um algoritmo que retorne uma solução em um curto intervalo de tempo. Uma das estratégias eficazes para resolver problemas combinatórios, como o roteamento de veículos é utilizar a meta heurística algoritmo genético.

Diante do exposto, o presente trabalho propõe desenvolver um sistema com a funcionalidade de encontrar a melhor rota entre uma origem e vários destinos utilizada restrições de capacidade do veículo, custo da sua utilização e janela de tempo no atendimento dos clientes. O objetivo da função de otimização de rota é de minimizar o custo do transporte, para operações de coleta de mercadorias. Para buscar a melhor rota será utilizado a meta heurística algoritmo genético.

A origem e os destinos serão obtidos através do Código de Endereçamento Postal (CEP), que serão convertidos para latitude e longitude, para poder calcular a distância e o tempo entre os pontos e representar em um mapa a rota calculada. Será utilizado o Bing Maps da Microsoft para retornar a latitude e longitude e calcular a distância e o tempo.

1.1 OBJETIVOS DO TRABALHO

O objetivo deste trabalho é disponibilizar um sistema de roteirização de veículos para operação de coletas utilizando algoritmos genéticos.

Os objetivos específicos podem ser detalhados da seguinte forma:

- a) calcular a melhor rota partindo de uma origem para um ou mais destinos com restrições de tempo;
- b) descobrir a distância entre dois pontos;
- c) identificar os veículos que irão compor a rota. Será verificada a capacidade de coletas das mercadorias e os veículos mais econômicos de uma determina frota;
- d) visualizar as rotas de coletas através de mapas.

1.2 ESTRUTURA DO TRABALHO

Este trabalho está organizado em 4 capítulos: introdução, fundamentação teórica, desenvolvimento e conclusão.

O segundo capítulo contempla a fundamentação teórica, onde são apresentados conceitos, sobre o problema de roteirização de veículos, técnicas de resoluções, tecnologias e trabalhos correlatos.

No terceiro capítulo são descritos os requisitos, a especificação do sistema e detalhes da implementação.

Por fim, são apresentados os resultados obtidos. No último capítulo são apresentadas conclusões e sugestões para futuros trabalhos.

2 FUNDAMENTAÇÃO TEÓRICA

A seção 2.1 descreve sobre a importância da logística. Na seção 2.2 problema da roteirização de veículos. Nas próximas seções são introduzidas estratégias conhecidas para resolução do problema de roteirização de veículo. Por fim, é apresentado o Bing Maps e a na última seção os trabalhos correlatos.

2.1 LOGÍSTICA

Segundo Moura (2006, p. 15) logística é o processo que gera a gestão dos produtos, de serviços e da informação, associada entre fornecedores, clientes, prestadores de serviços e outras entidades. Disponibilizando nas melhores condições produtos e serviços que incorporam valores, aos clientes aonde quer que eles estejam. Pode-se afirmar, junto com Moura (2006, p. 18) que a logística tem importância crítica na qualidade da vida da humanidade, pois ela disponibiliza produtos e serviços que podem ser desde máquinas, alimentos, roupa, livros entre outros, no momento que os consumidores precisam.

Nos tempos atuais a logística está sendo considerada uma estratégia chave para o sucesso atual das empresas no mercado corporativo, pois ela liga o mercado com os processo da organização (MOURA, 2006, p. 23). Em geral tem o objetivo de reduzir os custos operacionais, aumento de produtividade e melhora nos serviços oferecidos para os clientes

O sistema logístico é formado por vários canais de estratégias, entre eles há os canais de distribuição física, movimentação de matérias, armazenagem, gestão de estoque, tecnologia da informação, entre outras. Na figura 1, pode-se observar a visão sistêmica da cadeia de abastecimento, que engloba os canais de movimentação de matérias, armazenagem, distribuição física, transporte e gestão de estoque.

Figura 1 – Cadeia de abastecimento



Fonte: Guia do transportador (2014).

Observando a figura 1 pode-se notar que a logística possui três grandes processos que são o de suprimento físico, produção e distribuição física (GOMES, 2004, p. 9).

O processo de suprimentos físico (também é conhecido como gerenciamento de matérias) é o processo inicial da cadeia alimentar da logística. Este processo se encarrega de retirar a matéria-prima da sua origem para transportar até o local de manufatura (RIBEIRO, 2001, p. 16).

Segundo Ballou (2001), distribuição física é o processo que tem a responsabilidade de movimentar, estocar e efetuar o planejamento da distribuição dos pedidos finais até chegar no cliente final. Geralmente, neste processo também é definido os modais que serão usados e as rotas que serão feitas para entregar o produto para o cliente.

As etapas inerentes dos processos de suprimentos e a distribuição física são praticamente os mesmos, o que diferencia as duas etapas é que na logística de suprimentos trabalha com matérias-primas e a distribuição física trabalha com o produto acabado (GOMES, 2004, p. 27). Portanto, pode-se dizer que a logística tem como atividade primária a gestão de transporte, manutenção de estoque e processamentos de pedidos (GOMES, 2004, p. 9).

2.1.1 Gestão de Transporte

Um dos canais mais importantes na cadeia de abastecimento é o transporte, no qual exigem técnicas modernas de acondicionamento, manuseio, estocagem, transferência e movimentação da mercadoria (BARAT, 2007, p. 20). O custo do transporte é uma das maiores parcelas dos custos logísticos das empresas, variando de 4% até 25% do seu faturamento bruto, no Brasil estima-se que estes custos estão na ordem de 60 bilhões (Gomes, 2004, p. 67).

Segundo Gomes (2004, p. 70), uma das variáveis que é usada para calcular o preço de um produto logístico é o custo do valor do frete. Portanto, um sistema de transporte adequado é uma estratégia importante para ajudar a reduzir preços de mercadorias e aumentar a competitividade.

Segundo Gomes (2004, p. 67), no Brasil há maior parte do fluxo, entorno de 70% das movimentações de cargas é pelo modal rodoviário. Neste modal, os custos são separados em direto que ocupa cerca de 85% do custo total e o custo indireto que ocupa cerca de 15%. O custo direto está relacionado com a operação e a depreciação do veículo, remuneração do capital, salário e gratificações dos motoristas e ajudantes, combustível, lubrificação, pneus e licenciamento. Já o custo indireto está ligado ao setor de administração da empresa em termos gerais, que passa pela contabilidade, canal de vendas, diretoria, financeiro entre outros (GOMES, 2004, p. 72).

2.2 PROBLEMA DE ROTEIRIZAÇÃO DE VEÍCULO (PRV)

Segundo Miura (2003, p. 30) o problema de roteirização de veículo tem o objetivo de visitar vários clientes e encontrar a melhor rota respeitando determinadas restrições de um problema específico. Este tipo de problema está presente em empresas do segmento de transporte e distribuição onde muitas vezes essa é a sua principal atividade e é necessário ter uma boa estratégia para minimizar os custos com o transporte.

A solução proposta por Dantzig e Ramser (1959 apud ASSIS, 2007, p. 19) possui as seguintes restrições básicas:

- a) cada cidade é visitada uma única vez por um único veículo;
- b) cada rota é iniciada num depósito e finalizada no mesmo depósito;
- c) todas as demandas de todos os consumidores devem ser satisfeitas.

Entretanto podem ser adicionadas outras restrições como:

- a) restrição de janela de tempo: cada ponto deve ser visitado em um período de tempo específico (ASSIS, 2007, p. 6);

- b) capacidade dos veículos: veículos com capacidade limitada ou ilimitada (ASSIS, 2007, p. 6).

À medida que aumenta a quantidade de clientes a ser atendidos, a solução para buscar a melhor rota fica mais complexa do ponto de vista computacional, pois o número de combinações vai se tornando cada vez maior (SENS, 2009, p. 16). Essa explosão combinatória gera um tempo de processamento inviável por processo normal, tornando a aplicação inviável nestes casos.

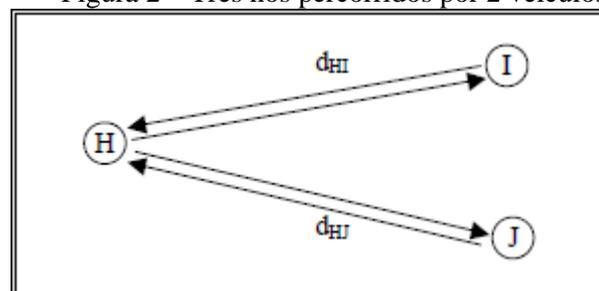
O problema de PRV como quase todo problema de análise combinatória é classificado como NP-Completo (SENS, 2009, p. 15), o que significa que a ordem de complexidade do esforço computacional aumenta exponencialmente conforme a quantidade de pontos a serem atendidos (GAREY; JOHNSON, 1999 apud MIURA, 2003, p. 31).

Podem ser usados três tipos de algoritmos para tentar encontrar uma solução para o problema: os algoritmos exatos, as heurísticas construtivas e as heurísticas baseadas em meta-heurísticas (ASSIS, 2007, p. 14). Segundo Miura (2003, p. 23), nos casos que há poucas restrições são usados os algoritmos exatos com técnicas de relaxações para resolver o problema de restrições e nos casos com muitas variáveis e restrições são usadas às heurísticas que buscam encontrar uma solução próxima da ótima.

2.3 CLARKE E WRIGHT

O algoritmo Clarke e Wright é uma das soluções clássicas para resolução do problema de roteamento de veículos com restrições de janela de tempo e capacidade do veículo. Ele é uma heurística do tipo economia, que busca substituir os arcos mais caros de um grafo dentro da rota por arcos de menores custos de forma a ser criada uma rota melhorada (MIURA, 2003, p. 31). Para entender a função da economia pode-se analisar as Figuras 2 e 3.

Figura 2 – Três nós percorridos por 2 veículos

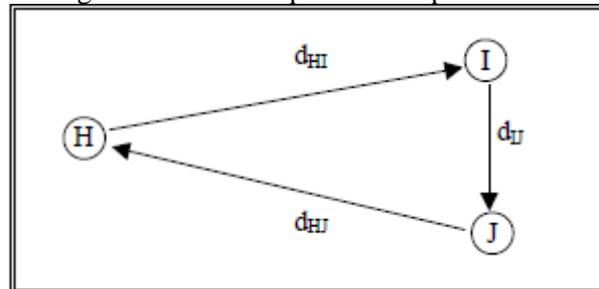


Fonte: Miura (2003, p.33).

Na figura 2 o H representa um depósito onde pode ter vários veículos disponíveis e o I e o J representam pontos de coletas. Na figura 2 dois veículos saem do depósito para coletar a mercadoria. Cada um é responsável por ir até um ponto de coleta e voltar para o depósito.

Já na figura 3 é eliminado um veículo do roteiro, fazendo que apenas um veículo percorra os três nós do problema, assim gerando uma economia na distância percorrida da rota.

Figura 3 – Três nós percorridos por um veículo



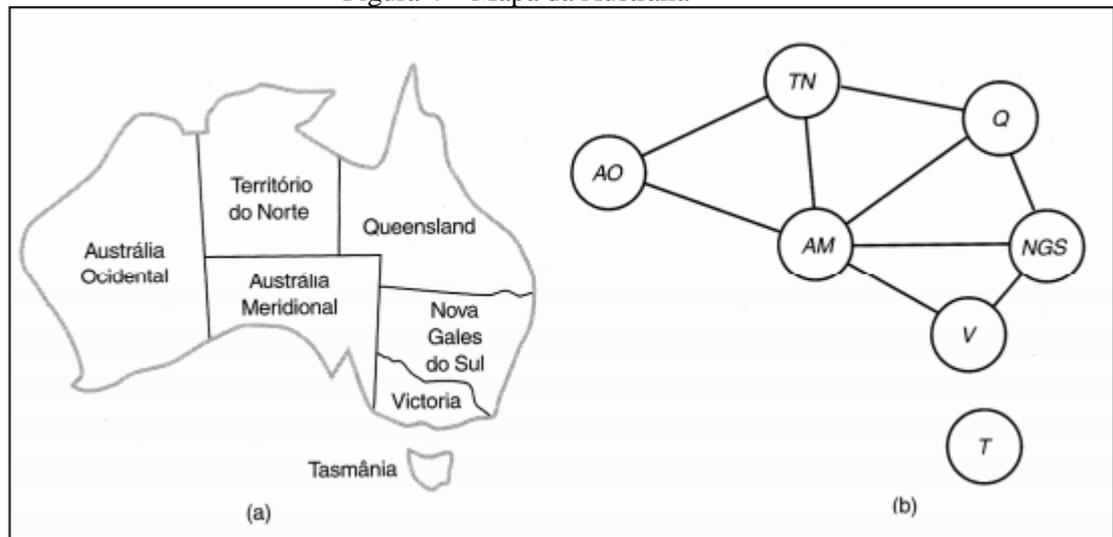
Fonte: Miura (2003, p.33).

2.4 CONSTRAINT SATISFACTION PROBLEM (CSP)

Segundo Sucupira (2003, p. 1), programação por restrição é uma tecnologia que tem como a sua principal característica a modelagem do problema. A modelagem consiste em especificar um conjunto de variáveis, o domínio de cada uma delas e o conjunto de restrições adicionais, estabelecendo as relações entre as variáveis. Formalmente, de acordo com Russel e Norvig (2004, p. 135), pode-se definir CSP como um conjunto de variáveis V_1, V_2, \dots, V_n e um conjunto de restrições R_1, R_2, \dots, R_m , onde cada variável V_1 tem um domínio D_1 que não pode ser vazio e cada restrição R_1 tem um subconjunto de variáveis V_1 que especificam as combinações de valores permitidas para aquele subconjunto.

Um exemplo prático apresentado por Russel e Norvig (2004, p. 138) é o problema de coloração de mapas, onde deseja colorir com um número definido de cores um mapa com regiões definidas e finitas, possuindo determinadas restrições. Na Figura 4 (a) é mostrado o mapa da Austrália que precisa ser pintado e na Figura 4 (b), um grafo, representando as fronteiras das regiões.

Figura 4 – Mapa da Austrália



Fonte: Russel e Norving (2004, p. 138).

Uma solução possível para este problema e a sua modelagem em CSP é apresentado no Quadro 1.

Quadro 1 – Definição e Solução do problema de coloração de mapas modelado em CSP.

<pre> Variáveis = { AO, TN, Q, NGS, V, AM, T } Restrições = { (AO ≠ TN), (AO ≠ AM), (TN ≠ AO), (TN ≠ AM), (TN ≠ Q) ..., (NGS ≠ V) } Domínio para as variáveis = { vermelho, verde, azul } Solução possível = { (AO = vermelho), (TN = verde), (Q = vermelho), (NGS = verde), (V = vermelho), (AM = azul), (T = vermelho) } </pre>
--

Fonte: Pires (2006, p. 16).

Na modelagem do problema apresentado no quadro 1, tem as variáveis que são os estados do mapa, o domínio de variáveis são as cores. A restrição é que cada região vizinha não pode possuir a mesma cor.

2.5 ALGORITMOS GENÉTICOS

Proposto por Holland (1975 apud PRESTES, 2006, p. 28), os algoritmos genéticos são uma meta-heurística inspirada na evolução dos processos biológicos. Baseia-se na teoria da evolução natural proposta por Charles Darwin (1959 apud PRESTES, 2006, p. 28), que sustenta a ideia que quanto melhor um indivíduo adapta-se ao seu meio ambiente, maior será

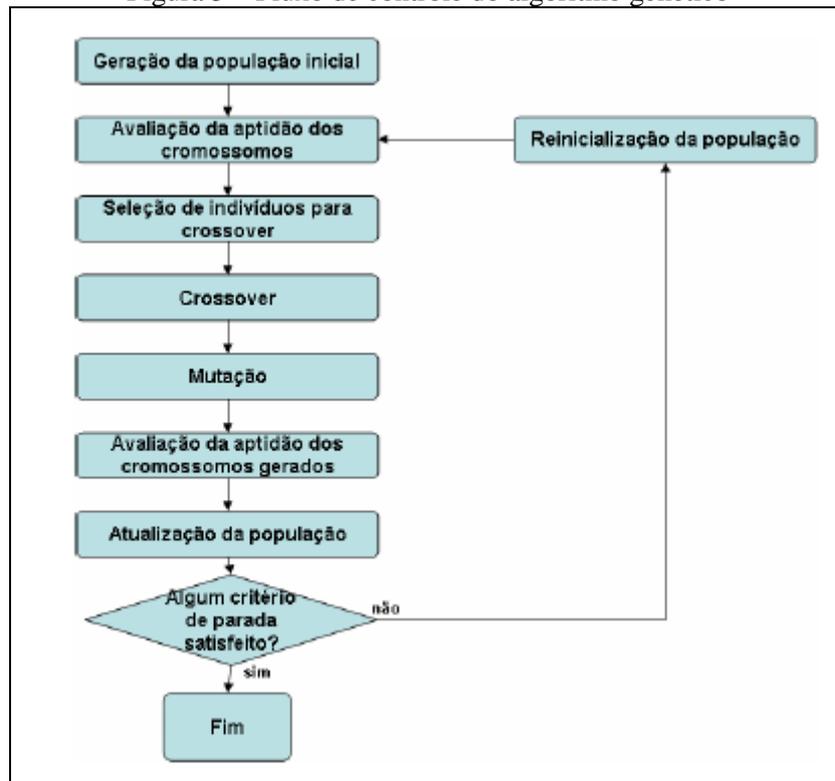
sua chance de sobreviver e gerar descendentes. A propagação destes indivíduos que perduram a espécie pode gerar um novo indivíduo mais adaptado ao seu meio.

Os algoritmos genéticos são apropriados para problemas complexos classificados como *NP-hard* e otimização combinatórios em geral, onde envolve muitas variáveis e espaços de soluções de dimensões elevadas. Segundo Malaquias (2006, p. 41), ultimamente o algoritmo genético está sendo aplicado em vários problemas *NP-hard* de domínios diversos, como a biologia, química, análise criptográfica, identificação de sistemas, medicina, microeletrônica, reconhecimento de padrões, planejamento da produção, robótica, entre outros.

Em linhas gerais, o funcionamento do algoritmo genético trabalha com uma população de indivíduos e cada indivíduo é representado por um cromossomo que é composto por genes, que são responsáveis pelas características dos indivíduos (ARAÚJO, 2008, p. 22).

Superficialmente o fluxo de um algoritmo genético pode ser visualizado na figura 5. Inicialmente é escolhida e gerada uma população inicial e determinada a sua quantidade. Em seguida a cada iteração são escolhidos cromossomos pais que irão passar pelas operações de *crossover* onde serão cruzados seus genes e a probabilidade de mutação que altera aleatoriamente o gene do novo cromossomo. Os indivíduos mais aptos a transmitir seus genes são adicionados na população e os menos aptos são eliminados, conforme uma avaliação de aptidão específica de um problema.

Figura 5 – Fluxo de controle do algoritmo genético



Fonte: Araújo (2008, p. 23).

2.5.1 Representação Cromossomial

A representação cromossomial basicamente é uma forma de traduzir o problema em questão em uma forma viável de ser tratada pelo computador, cada parte indivisível do cromossomo é um gene (LINDEN, 2008, p. 59). O cromossomo pode ser representado, por exemplo, por um tipo de dado binário, inteiro e real. Segundo Malaquias (2008, p. 45) a representação binária é o tipo clássico de representação de um cromossomo, por ser interpretada de uma forma mais fácil e por se encaixar melhor nas operações de *crossover* e *mutação*.

2.5.2 População

A população de um algoritmo genético é um conjunto de indivíduos que pode ser considerado solução de um determinado problema. Ou seja o espaço de busca que o algoritmo genético irá trabalhar para encontrar a melhor solução é chamado de população (MALAQUIAS, 2006, p. 48). A população inicial é gerada na maioria das vezes de forma simples, fazendo uma escolha aleatória independente para cada indivíduo da população (LINDEN, 2008, p. 64).

A cada nova geração da população todos os novos indivíduos são avaliados por uma função de aptidão que irá retornar uma pontuação para cada cromossomo. A função de

avaliação do cromossomo segundo Lucas (2002, p 11) avalia cada requisito desejável e soma em uma pontuação final para cada cromossomo.

2.5.3 Elitismo

A cada nova geração existe possibilidade de perder os melhores indivíduos e gerar uma população pior que a população atual (MALAQUIAS, 2006, p. 67). Segundo Linden (2008, p. 129), a ideia por trás do elitismo é que os N melhores indivíduos de cada geração deve ser passado para a próxima geração com a finalidade de ter os seus genes preservados.

2.5.4 Seleção de Cromossomo

Segundo Linden (2008, p. 67), o método que seleciona os indivíduos que irão participar da próxima geração devem simular a teoria da seleção natural que atua sobre as espécies biológicas. Nesse mecanismo os pais mais aptos geram mais filhos, ao mesmo tempo em que os pais menos aptos também podem gerar seus descendentes, essa característica na seleção é necessária para poder gerar soluções diversificadas e dar continuidade na evolução.

Na figura 6 tem-se um exemplo de população apresentada por Linden (2008, p. 68) e que usa uma função objetiva para alcançar o valor de 1111 de um determinado indivíduo. Como pode ser observado essa população só irá conseguir evoluir para um indivíduo com valor maior que 0110 se utilizarem o indivíduos com piores avaliação.

Figura 6 – Exemplo de população de indivíduos

<i>Indivíduo</i>	<i>Avaliação</i>
0001	1
0011	9
0100	16
0110	36
<i>Total</i>	<i>62</i>

Fonte: Linden (2008, p. 68).

2.5.5 Cruzamento

Segundo Lobo (2005, p. 55) o operador *crossover* ou cruzamento tem a finalidade de gerar novos indivíduos a partir de cromossomos selecionados da geração atual. A operação é realizada através da escolha de dois pares de indivíduos, que serão os pais e que terão seus genes combinados, gerando dois novos indivíduos filhos que serão diferentes dos pais mais com matéria genética de ambos (MALAQUIAS, 2006, p. 65).

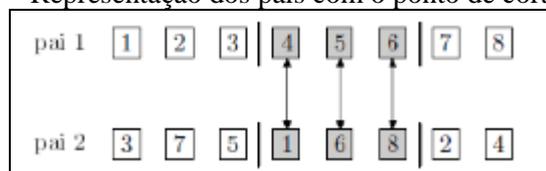
Com a utilização do operador de *crossover* espera aumentar a qualidade média da população, gerando soluções com maior qualidade para o problema em questão. Contudo é necessário aplicar um operador que define a probabilidade de cruzamento do par de cromossomos selecionados, para não deixar a estrutura mais homogênea a cada nova geração (Lopes, 1995, p. 35).

2.5.5.1 Cruzamento de mapeamento parcial PMX

O operador de cruzamento parcial (PMX) tem a finalidade de trocar uma parte dos genes de um pai para outro pai, sempre preservando a outra parte dos gene do pai no filho (MALAQUIAS, 2006, p. 75).

Um exemplo prático apresentado por Malaquias (2006, p. 75), considerando os pais com os genes (1 2 3 4 5 6 7 8) e (3 7 5 1 6 8 2 4), o PMX inicialmente seleciona dois pontos de corte no qual servirá de fronteira para trocar os genes dos pais. Hipoteticamente suponhamos que foi sorteado o primeiro ponto de corte entre o terceiro e quarto gene e o segundo entre o sexto e sétimo. Observando a figura 7 podem-se observar os pais com o seus pontos de corte que é representado pelo “|”. Onde as seções de mapeamentos dos pais são 4 -> 1, 5 -> 6 e 6 -> 8.

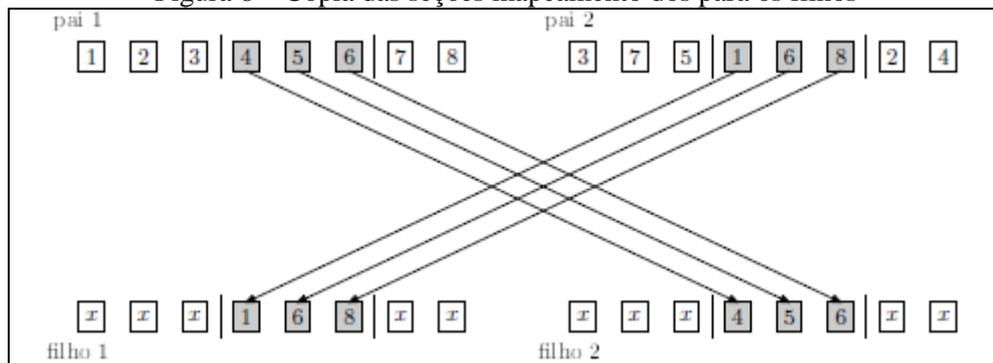
Figura 7 – Representação dos pais com o ponto de corte definido



Fonte: Malaquias (2006, p. 75).

O próximo passo do PMX é copiar os genes mapeados do pai 1 para o filho2 e do pai 2 para o filho1. A figura 8 demonstra as cópias dos genes dos pais para o filho.

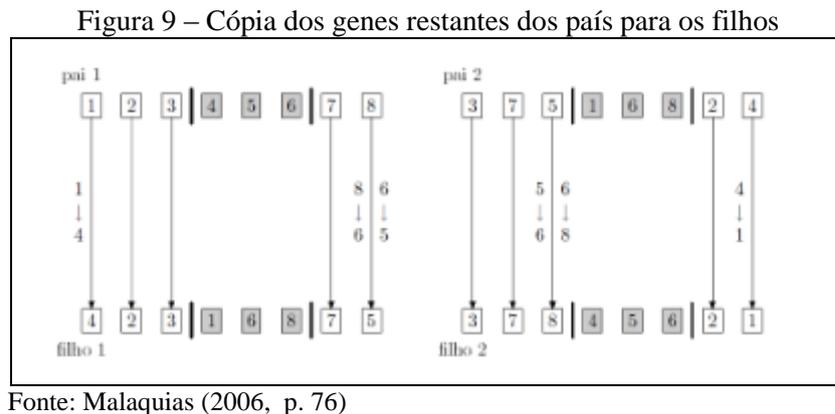
Figura 8 – Cópia das seções mapeamento dos para os filhos



Fonte: Malaquias (2006, p. 76).

O último passo do algoritmo é copiar os genes restantes do pai 1 para o filho 1 e do pai 2 para o filho 2. Caso já exista um gene do pai 1 no cromossomo do filho 1 é verificado na seção de mapeamento o gene que deve ser substituído. Exemplo, caso tente inserir o gene 1 do pai 1 no filho 1, deverá olhar a seção de mapeamento, pois esse gene já faz parte do cromossomo. Verificando a seção de mapeamento encontra o mapeamento do 1 -> 4, então troca o gene 1 pelo 4.

Esse processo ocorre também com o pai 2 e filho 2. E pode ser observado na figura 9.

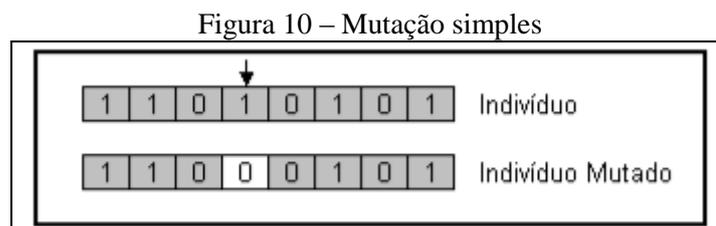


2.5.6 Mutação

Em cada nova geração existe a possibilidade do algoritmo genético ficar explorando apenas determinada região da geração anterior, o que pode ser um problema caso a região seja em uma área de mínimos locais (MALAQUIAS, p. 67).

Segundo Linden (2008, p. 129) o operador de mutação é uma heurística exploratória, que permite buscar soluções fora dos limites definidos na população inicial. Esta operação altera características de um indivíduo de forma aleatória conforme uma taxa de mutação.

Um exemplo de mutação em uma estrutura binária pode ser visto na figura 10, nela é trocado um bit 1 para um bit 0.



2.6 BING MAPS

Segundo Boeck (2012, p. 23), Bing Maps é um serviço online de mapas e satélite de forma interativa, que permite o usuário pesquisar, compartilhar, descobrir e planejar informações sobre determinados locais, usando mapas rodoviários, vistas fotográficas de altas resoluções e efetuar buscas por aproximações.

A Microsoft disponibiliza um conjunto de API (*Application Programming Interface*) que são compatíveis com plataformas específicas e possui serviços específicos.

Entre as opções de API's destacam-se (Microsoft, 2014):

- a) Bing Maps Applications for Windows 8: permite integrar o Bing Maps com aplicativos para o Windows 8. Alguns serviços disponíveis são: obter dados de geocodificação, obter dados de trânsito e trabalhar com mapas;
- b) Bing Maps Rest: utiliza os princípios RestFull para fazer a comunicação com os serviços, portanto pode ser executado em qualquer ambiente. Alguns dos serviços disponíveis são: calcular a distância entre dois pontos, obter geocodificação e obter dados de trânsito;
- c) Bing Maps WCF: integra uma aplicação Windows Presentation Foundation (WCF) com os serviços de Mapas do Bing;
- d) Bing Maps AJAX V7: é a API mais universal para ser usada na comunicação com os serviços do Bing, sendo a mais indicada para o desenvolvimento de aplicação Web. Com ela se pode trabalhar com mapas, obter dados de direções e trânsito e obter geocodificação.

Para obter acesso a esses recursos é necessário criar uma chave de validação junto a Microsoft.

2.7 TRABALHOS CORRELATOS

Existem vários trabalhos acadêmicos e comerciais que abordam o problema de roteirização. Os softwares de mercado não disponibilizam especificação das técnicas que é utilizado para resolução do problema. Portanto, os trabalhos correlatos abordados são Roteirização de veículo (MIURA, 2003). Sistema para logística de distribuição (ROSA, 2005) e Modelo de roteirização de veículos (TSUDA, 2007).

2.7.1 Roteirização de veículo

O trabalho de Miura (2003) aborda como sua principal preocupação melhorar a coleta de mercadorias de uma empresa transportadora de grande porte através de métodos

científicos, tendo como objetivo a redução de custos na geração das rotas de coleta de mercadorias. Para calcular a rota foram utilizadas restrições de janela de tempo, restrições relacionadas à capacidade do veículo e o tempo de expediente do motorista.

Para atingir o objetivo foi utilizada uma adaptação da heurística Clarke e Wright que obteve sucesso para enquadrar a melhor rota. Segundo o autor a escolha pela a heurística a ser implementada no trabalho foi baseada no fato de Clarke e Wright apresentar bons resultados na obtenção de soluções para problemas de roteamento de veículos e permite adicionar diversas restrições.

Os objetivos do trabalho foram alcançados, gerando uma economia de 29,3% da distância percorrida pelos veículos. Os testes foram feitos com até 103 pontos de coletas na cidade de São Paulo e os pontos foram divididos em 7 regiões de zoneamento da cidade de São Paulo, que teve a roteirização processada paralelamente para cada região.

De acordo com Miura (2003, p. 31), Clarke e Wright é definido como um modelo heurística de economia que busca substituir arcos mais caros dentro da rota por arcos de menor custos, criando uma rota melhorada.

2.7.2 Sistema para logística de distribuição

O trabalho de Rosa (2005) tem o objetivo de construir um sistema de logística e de distribuição com o objetivo de otimizar as rotas de entrega e coleta de mercadoria através da heurística *Constraint Satisfaction Problem (CSP)*.

Os objetivos específicos do trabalho são: conseguir sugerir quais produtos serão transportados por determinados veículos e conseguir definir a melhor rota para o produto chegar no determinado destino. Os objetivos do trabalho foram parcialmente alcançados, pois o sistema conseguiu encontrar a melhor rota de forma eficiente, contudo a heurística não conseguiu determinar o veículo que irá fazer parte de uma rota. Os testes do trabalho foram feitos com até 26 variáveis e demorou cerca de 1 hora para encontrar o melhor caminho.

No sistema desenvolvido é necessário informar a distância de forma manual pelo usuário entre uma origem e um destino. Para calcular a rota foram utilizadas as restrições de controle do peso e volume da carga do veículo em cada entrega com a finalidade de não permitir que a capacidade de armazenagem do veículo seja ultrapassada e o custo de cada distância que o veículo vai percorrer com a finalidade de encontrar a melhor.

A linguagem de programação utilizada foi *Object Pascal*, o ambiente de desenvolvimento Delphi e *GNU Prolog* para compilar o algoritmo de CSP desenvolvido na linguagem *Prolog*.

2.7.3 Modelo de roteirização de veículos

Este trabalho tem o objetivo de aplicar técnicas da pesquisa operacional para resolver problemas de roteirização de veículos na operação de entregas de mercadorias. O problema que se deseja resolver consiste em determinar a melhor maneira de alocar os veículos para realizar os roteiros diários de entrega, com a finalidade de minimizar os custos de transporte e diminuir a distância percorrida.

O modelo desenvolvido foi baseado em resolver alguns problemas de uma empresa de importação de produtos Japoneses. Onde foi possível testar a solução em cenários reais de testes. Este modelo foi desenvolvido de forma simplificada, em relação as todas restrições que a empresa possui para roteirizar. Foi implementada a restrição de respeitar a capacidade máxima de carga do veículo.

Para resolver o problema foi utilizado o algoritmo de varredura *sweep algorithm*. Este algoritmo Foi adaptado para ser executado no ambiente Excel.

Os resultados obtidos no diversos cenários de testes constataram que a solução conseguiu reduzir uma economia média de 18,24% na distância percorrida, comparando com a situação atual de entrega da empresa estudada.

3 DESENVOLVIMENTO

Este capítulo detalha as etapas do desenvolvimento do sistema. São apresentados os requisitos, a especificação e a implementação, mencionando as técnicas e ferramentas utilizadas. Também é apresentada a operacionalidade do software e os resultados obtidos.

3.1 REQUISITOS PRINCIPAIS DO PROBLEMA A SER TRABALHADO

A partir do estudo dos trabalhos correlatos, podem-se definir os requisitos da ferramenta, que deve possuir um módulo de cadastros, o qual reunirá as informações necessárias para a roteirização e um módulo de operação, onde o usuário poderá visualizar a rota. A seguir são apresentados os Requisitos Funcionais (RF) e Requisitos Não Funcionais (RNF):

- a) ser desenvolvido na linguagem C# (RNF);
- b) utilizar o Microsoft Visual Studio 2010 ou superior como ambiente de desenvolvimento (RNF);
- c) utilizar *web services* para disponibilizar o serviço que busca a melhor rota (RNF);
- d) utilizar o banco de dados SQL Server 2008 ou superior (RNF);
- e) calcular a melhor rota entre um ponto de origem e um ou mais pontos de destinos (RF);
- f) calcular a distância em quilômetros (km) entre uma origem e um destino (RF);
- g) buscar a longitude e latitude dos clientes e armazenar-das no banco caso não exista cadastrada (RF);
- h) identificar os veículos que irão fazer determinada rota (RF);
- i) permitir cadastrar veículos (RF);
- j) permitir cadastrar clientes (RF);
- k) permitir cadastrar pontos de coleta (RF);
- l) permitir alterar configurações do algoritmo genético (RF);
- m) disponibilizar uma interface para visualização da rota (RF);
- n) permitir cadastrar restrições dos problemas (RF).

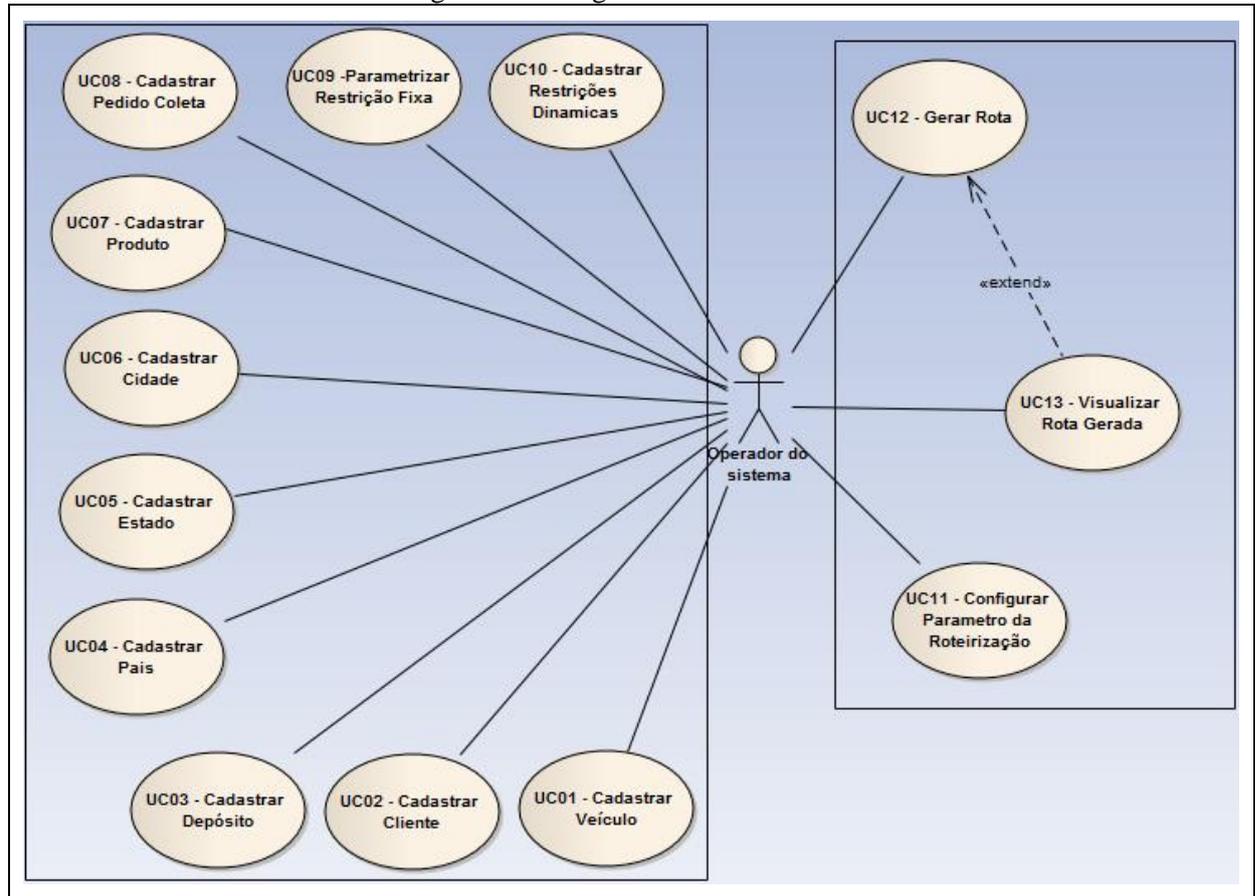
3.2 ESPECIFICAÇÃO

Nesta seção é apresentada a especificação do sistema. A especificação foi desenvolvida seguindo a análise orientada objeto utilizado *Unified Modeling Language* (UML). Os diagramas foram desenvolvidos na ferramenta Enterprise Architect 7.5.

3.2.1 Diagrama de Casos de Uso

Na (Figura 11) está representando o caso de uso principal do sistema, as ações que o operador do sistema deve possuir para a execução do sistema. Estas ações são necessárias para gerar rota.

Figura 11 – Diagramas de caso de uso



Os casos de usos são:

- UC01 – cadastrar veículo: permite incluir, alterar, consultar e excluir um veículo;
- UC02 – cadastrar cliente: permite incluir, alterar, consultar e excluir um cliente;
- UC03 – cadastrar depósito: permite incluir, alterar, consultar e excluir um depósito;
- UC04 - cadastrar país: permite incluir, alterar, consultar e excluir um país;
- UC05 – cadastrar estado: permite incluir, alterar, consultar e excluir um estado;
- UC06 – cadastrar cidade: permite incluir, alterar, consultar e excluir uma cidade;

- g) UC07 – cadastrar produto: permite incluir, alterar, consultar e excluir um produto;
- h) UC08 – cadastrar pedido de coleta: permite incluir um pedido para ser coletado e especificar os produtos que compõe o pedido;
- i) UC09 – parametrizar restrição fixa: permite parametrizar se o sistema irá usar as restrições de Janela de tempo, capacidade de veículo e restrições dinâmicas;
- j) UC10 – cadastrar restrição dinâmica: permite cadastrar novas restrições para o sistema;
- k) UC11 – configurar parâmetro roteirização: permite alterar configurações do algoritmo genético;
- l) UC12 - gerar rota: permite gerar a rota para determinados pedidos de coletas de um depósito conforme um intervalo de datas e hora;
- m) UC13 – visualizar rota gerada: permite visualizar a rota gerada pelo sistema em uma mapa que utiliza a API do Bing Maps.

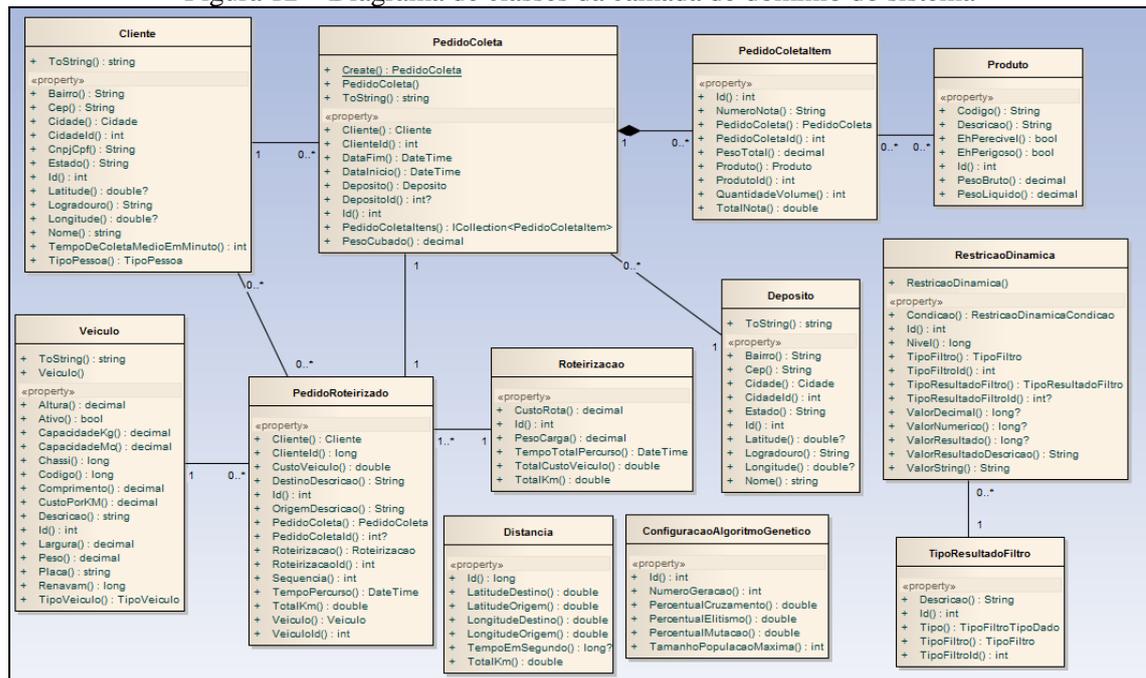
3.2.2 Diagrama de classes

Nesta seção são descritas as estruturas e classes do sistema. Os diagramas de classes definidos foram o da camada de domínio do sistema e o da camada *engine*, sendo mostradas as classes mais importantes destas camadas.

3.2.2.1 Diagrama de classes simplificado da camada de Domínio

Na Figura 12 é apresentado o diagrama de classes da camada de domínio que mostra como está estruturado o sistema.

Figura 12 – Diagrama de classes da camada de domínio do sistema



As classes dos sistemas apresentadas na figura 12 são:

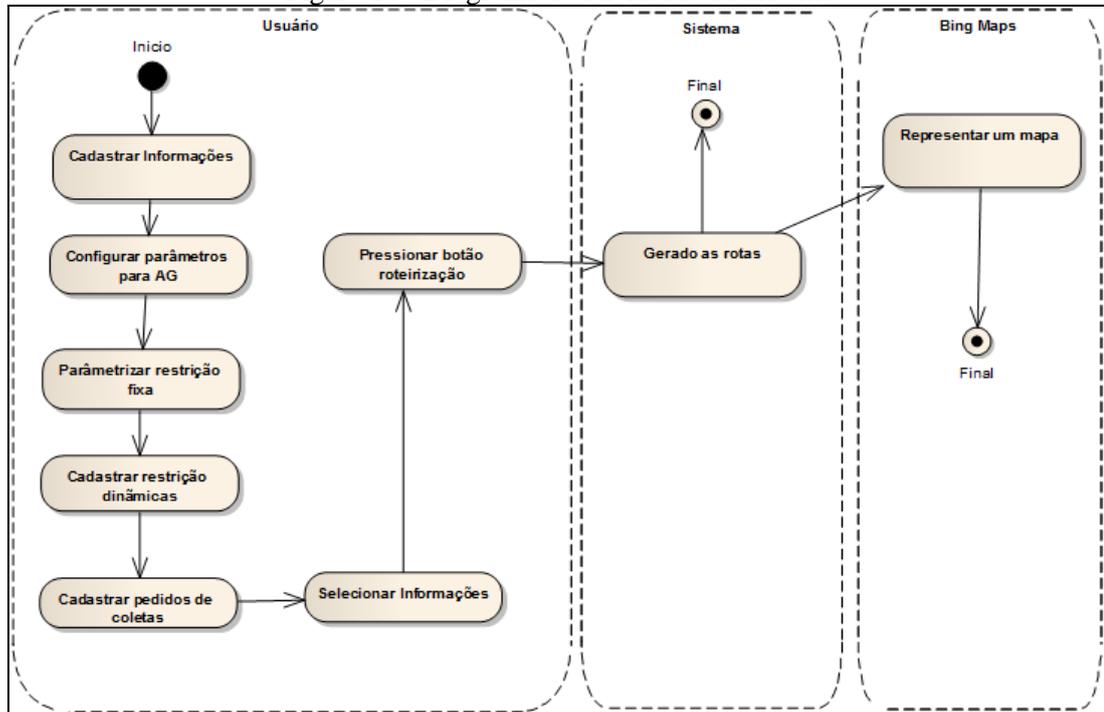
- Cliente: classe que representa um cliente com seus determinados atributos;
- PedidoColeta: classe responsável por representar um pedido de coleta realizado pelo cliente;
- ItemPedidoColeta: classe responsável por representar os itens de pedidos do pedido de coleta;
- Produto: classe que representa a definição de um produto;
- Roteirizacao: classe com a responsabilidade de representar uma roteirização contém informações de totalizadores da rota;
- PedidoRoteirizado: classe com a responsabilidade de representar os detalhes de cada roteiro da roteirização;
- Depósito: classe que representa um depósito no sistema;
- Distancia: para cada origem e destino há informações da distância e o tempo, portanto esta classe representa estas informações;
- Veiculo: classe que representa a definição de um veículo;
- ConfiguracaoAlgoritmoGenetico: representa os atributos de configuração do algoritmo genético implementados;
- RestricaoDinamica: define as restrições dinâmicas do sistema.

3.2.3 Diagrama de atividades

A seguir são apresentados os diagramas de atividades do fluxo do sistema e da rotina de roteirização.

O diagrama de atividades apresentado na Figura 19 ilustra os processos do sistema.

Figura 14 - Diagrama de atividades do sistema



O primeiro passo para o usuário é cadastrar todas as informações pertinentes a roteirização, como produtos, clientes, veículos, depósitos e endereços.

Em seguida é necessário configurar os parâmetros e restrições. Nas configurações do algoritmo genético é necessário configurar o percentual de mutação, percentual de cruzamento, entre outros. Na parte de parametrização é necessário informar se a roteirização irá utilizar as restrições de janela de tempo, capacidade do veículo e se irá utilizar restrição dinâmica.

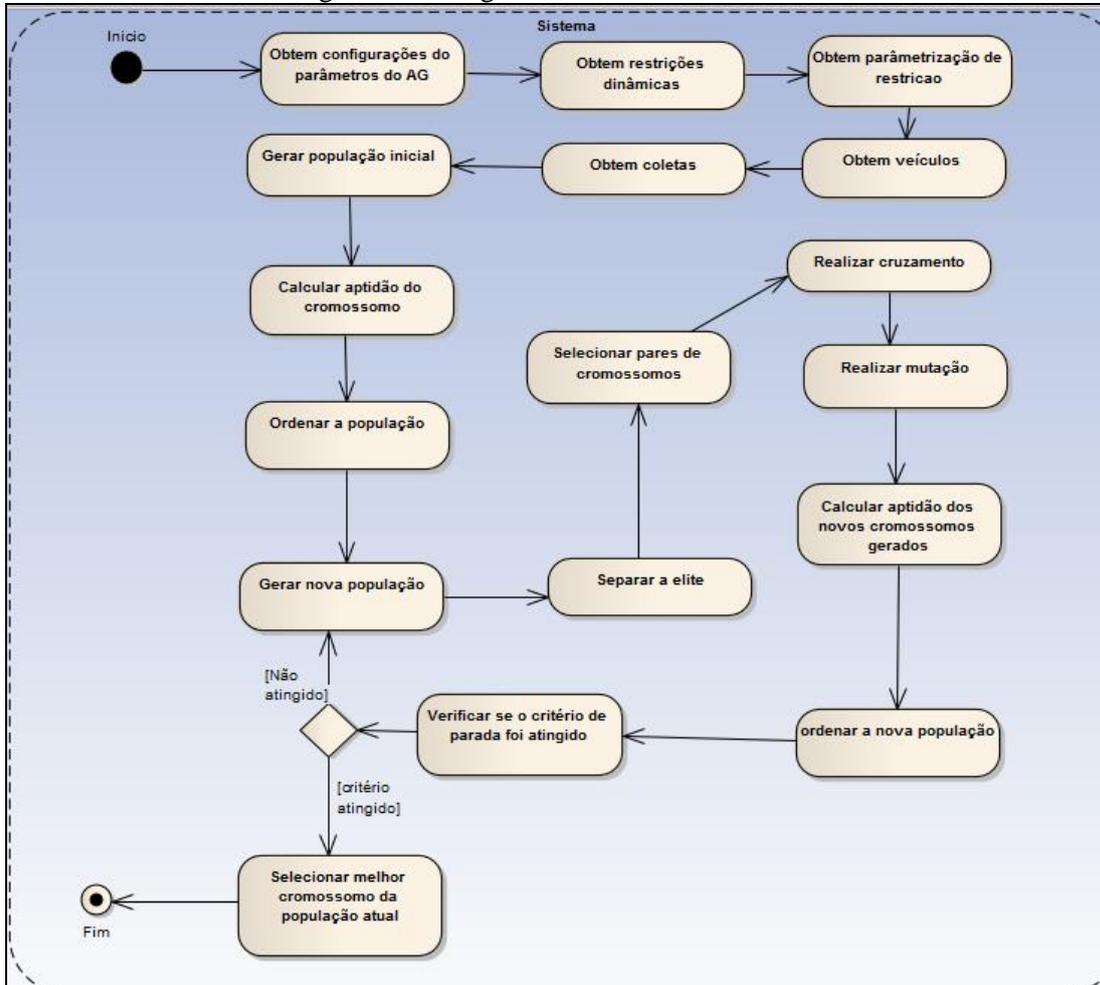
No passo seguinte o usuário pode montar condições dinâmicas com as entidades do sistema para serem usadas como restrições na rotina de roteirização.

No próximo passo o usuário precisa cadastrar os pedidos de coletas dos clientes que serão usados na roteirização.

No sistema é gerado uma ou mais rotas conforme as informações selecionadas pelo usuário, cadastros dos sistemas e os parâmetros configurados no sistema. As informações que o usuário precisa selecionar são pedidos de coletas, horário do início da coleta e a origem das coletas. Após isso o usuário pode visualizar determinada rota gerada em um mapa que utiliza a API do Bing Maps.

O diagrama apresentado na Figura 15 demonstra o funcionamento da rotina de roteirização.

Figura 15 – Diagrama de atividade da rotina roteirizar



3.2.3.1 Gerar população inicial

Inicialmente a rotina gera uma população inicial de cromossomos de forma aleatória, contudo é verificado se o cromossomo gerado é uma solução consistente, ou seja não pode ter genes de coletas repetidas em um cromossomo. O tamanho da população é definido na funcionalidade de configuração de parâmetro do roteirizador.

3.2.3.2 Ordenar população

Nesta etapa é ordenada a população atual de cromossomos. A ordenação é realizada com base na aptidão de cada cromossomos de forma ascendente, ou seja, os cromossomos com menor aptidão estarão no topo da população.

3.2.3.3 Calcular aptidão

O processo de calcular aptidão tem a responsabilidade de avaliar a qualidade de todos os cromossomos. Esta avaliação é feita através de uma função objetivo, que foi modelada com o objetivo de minimizar o custo de transporte na operação de coleta.

A função objetivo precisa considerar os objetivos do problema que se deseja resolver na sua fórmula. No caso do problema de roteirização de veículos estudado neste trabalho, possui as seguintes variáveis: veículos, pedidos de coletas, restrições dinâmicas, restrição de janela de tempo, restrição de capacidade do veículo. Pode-se visualizar a função quadro 2.

Quadro 2 - Fórmula

$$\text{Aptidão} = \sum ((\text{NPg} * (\text{DTrg} * (\text{DTrg} * \text{CPkmrg}))))$$

Onde:

Aptidão = é o custo do cromossomo

\sum = somatório de todos os genes

NPg = número de penalidades do gene

DTrg = distância da rota do gene

CPkmrg = custo do veículo por quilômetro na rota do gene

Para calcular a aptidão do cromossomo é necessário calcular todos os seus genes. Cada gene contém uma origem, destino e um veículo.

A variável número de penalidades do gene possui a finalidade de somar todas as restrições de Janela de tempo e restrições dinâmicas de uma determinada rota. Com a finalidade de deixar os cromossomos que possui restrições com uma aptidão alta. Por padrão a variável número de penalidades possui o valor 1.

A restrição de capacidade do veículo, não é contabilizada na variável número de penalidades, pois quando é inferida esta restrição a rotina manda o veículo de volta para o depósito para descarregar e retorna ao ponto que aconteceu a restrição, dobrando a variável km percorrido.

A variável distância da rota tem a responsabilidade de salvar a distância entre uma origem e um destino. Esta variável é muito importante na função para prevalecer as distâncias mais curtas. Para saber a distância em quilômetro entre uma origem e um destino é utilizado a API do Bing Maps.

O custo do veículo por quilômetro na rota, após calcular a distância da rota é multiplicado este valor com o valor do custo do veículo por quilômetro rodado. Esta informação de custo do veículo é registrado no cadastro do veículo. A intenção desta variável é dar prioridade para os veículos menos custosos.

3.2.3.4 Nova geração

O processo de nova geração tem a responsabilidade de gerar uma nova população com base na população da geração anterior. Para isso, são utilizadas algumas funções como a elitização dos cromossomos e a função dos operadores genéticos cruzamento e mutação.

3.2.3.5 Separar Elitização

Neste processo são preservados os melhores cromossomos da população atual para sobreviver na nova geração. Portanto, esta forma garante que a qualidade dos cromossomos gerados na nova população não seja pior que a anterior. A quantidade de cromossomos que irão sobreviver é definido através de um percentual de elitismo que é definido na funcionalidade de configuração do parâmetro da roteirização.

3.2.3.6 Operadores genéticos

Para gerar novos cromossomos são selecionados dois cromossomos aleatórios da população atual e verificado através da taxa de cruzamento se estes cromossomos terão os seus genes cruzados dando a origem há dois novos cromossomos. Caso, a taxa de cruzamento não permita o cruzamento é verificado se será feita a mutação dos cromossomos através da taxa de mutação.

A rotina de cruzamento desenvolvida é baseada nos princípios do operador PMX que foi apresentado no item 2.5.5.1. As informações da taxa de cruzamento e mutação podem ser alteradas na tela de configurar parâmetro roteirização.

3.2.3.7 Critério de parada

O critério de parada do algoritmo genético foi definido por uma quantidade de gerações que será gerada uma nova população. Após atingir o critério de parada é selecionado o melhor cromossomo da população atual.

A quantidade de gerações é definida na funcionalidade de configuração do parâmetro da roteirização.

3.3 IMPLEMENTAÇÃO

A seguir são mostradas as técnicas e ferramentas utilizadas e a operacionalidade da implementação.

3.3.1 Técnicas e ferramentas utilizadas

O sistema foi desenvolvido na linguagem C# / Asp.net e javaScript sobre a plataforma Microsoft .NET (versão 4.5). Para codificação foi usado o ambiente Microsoft Visual Studio 2012, também foram utilizados frameworks para agilizar o desenvolvimento e melhorar os resultados.

- a) ASP.NET MVC *framework*: é uma tecnologia para desenvolvimento na plataforma ASP.NET que utiliza o modelo arquitetural model, view, controller.
- b) Microsoft Entity *framework*: é uma tecnologia de acesso de dados.
- c) ASP.NET Web API *framework*: é uma tecnologia que facilita a criação de serviços rest http na plataforma .Net.

3.3.2 Desenvolvimento do sistema

Nesta seção são apresentados os detalhes da implementação das rotinas que compõe o sistema.

3.3.2.1 Calcular aptidão

Pode-se visualizar o trecho de código que calcula a aptidão de um cromossomo no quadro 3. É realizada uma iteração de todos os genes de determinado cromossomo e multiplicadas as penalidades encontradas, com o custo em km do percurso e o custo do veículo.

Quadro 3 – calcular aptidão do cromossomo

```
Double custoVeiculo =
(custoKMPercurso * Convert.ToDouble(veiculo.CustoPorKM));
aptidaoCromossomo +=
(numeroPenalidade * Convert.ToDouble(custoVeiculo));
```

3.3.2.2 Cruzamento

A rotina de cruzamento é baseada no algoritmo do operador PMX. Pode-se visualizar trechos importantes do seu código nos Quadros 4 e 5. A rotina recebe como parâmetro dois

cromossomos o pai1 e pai2. Primeiramente ela sorteia um intervalo de pontos que será percorrido e criados pontos de mapeamentos dos genes do pai1 para o pai2 e do pai2 para o pai1. Após estes genes mapeados são substituídos sempre preservando a consistência dos genes no cromossomo. Ou seja, não pode haver genes com coletas iguais no mesmo cromossomo.

Quadro 4 – Início da Rotina de cruzamento

```

IList<Gene> filhoGene1 = new List<Gene>(pai.Genes.ToList());
IList<Gene> filhoGene2 = new List<Gene>(pai2.Genes.ToList());

int pontoInicio;
int pontoFinal;
intervalorDeCorte(out pontoInicio, out pontoFinal);

Dictionary<int, Gene> pontoMapeamentoPai2ParaPai= new Dictionary<int,
Gene>();
Dictionary<int, Gene> pontoMapeamentoPaiParaPai2 = new Dictionary<int,
Gene>();

for (int i = pontoInicio; i <= pontoFinal; i++)
{
    pontoMapeamentoPai2ParaPai.Add
    (
        filhoGene2[i].PedidoColeta.Id, filhoGene1[i]
    );
    pontoMapeamentoPaiParaPai2.Add
    (filhoGene1[i].PedidoColeta.Id, filhoGene2[i]
    );
    Gene geneAlteradoFilho1 = filhoGene1[i];
    filhoGene1[i] = new Gene()
    { PedidoColeta = filhoGene2[i].PedidoColeta,
      Veiculo = filhoGene2[i].Veiculo
    };
    filhoGene2[i] = new Gene()
    {
        PedidoColeta = geneAlteradoFilho1.PedidoColeta,
        Veiculo = geneAlteradoFilho1.Veiculo
    };
}

```

Quadro 5 – Continuação da Rotina de cruzamento

```

for (int j = 0; j < filhoGenel.Count; j++)
{
    if (j < pontoInicio || j > pontoFinal)
    {
        if (pontoMapeamentoPai2ParaPai.ContainsKey(
            filhoGenel[j].PedidoColeta.Id))
        {
            Gene gene =
            pontoMapeamentoPai2ParaPai[filhoGenel[j].PedidoColeta.Id];
            while (pontoMapeamentoPai2ParaPai.ContainsKey(
                gene.PedidoColeta.Id))
            {
                gene = pontoMapeamentoPai2ParaPai[gene.PedidoColeta.Id];
            }
            filhoGenel[j] = new Gene()
            {
                PedidoColeta = gene.PedidoColeta, Veiculo = gene.Veiculo
            };
        }
    }
}

```

3.3.2.3 Mutação

No Quadro 6 é possível visualizar o método `swap()`, que é responsável por fazer a mutação de um cromossomo. Baseado na quantidade de genes que um cromossomo possui é sorteados dois índices, que são utilizados para inverter as posições dos genes.

Quadro 6 – rotina de mutação

```

internal void swap()
{
    Random random = new Random();
    int index1 = random.Next(quantidadeDeGenes());
    int index2 = random.Next(quantidadeDeGenes());

    if (index1 != index2)
    {
        Gene temp = new Gene(this.Genes[index1].Veiculo,
            this.Genes[index1].PedidoColeta);
        this.Genes[index1] = new Gene(
            this.Genes[index2].Veiculo, this.Genes[index2].PedidoColeta);
        this.Genes[index2] = new Gene(temp.Veiculo, temp.PedidoColeta);
    }
}

```

3.3.2.4 Calcular distância e tempo entre uma origem e um destino

No quadro 7 é mostrado o método `custosDoKmETempoEntreDoisPontos()`. Esta rotina recebe como parâmetro um objeto distância que contém as informações de latitude e longitude entre os pontos. Após, é pesquisado e retornado estas informações. Todas as requisições feitas para o serviço da API do Bing que calcula a distância em quilômetros e o

tempo são salvas no banco de dados e carregadas no início do sistema. Portanto, sempre é verificado se esta informação já não foi calculada, antes de chamar o serviço do Bing.

Quadro 7 – Método que calcula o custo em km e Tempo entre dois pontos

```

Distancia distanciaRetorno = null;
string chave = distancia.LongitudeOrigem + ";" +
distancia.LatitudeOrigem + ";" + distancia.LongitudeDestino + ";" +
distancia.LatitudeDestino;
if (distanciasCalculadas.ContainsKey(chave))
{
    distanciaRetorno = distanciasCalculadas[chave];
}
else
{
    Distancia novaDistancia = new Distancia();
    try
    {
        string distanciaCalculada =
newControleLongitudeLatitude().CalcularQuilometroETempoEntreDoisPonto
(
    distancia.LatitudeOrigem, distancia.LongitudeOrigem,
    distancia.LatitudeDestino, distancia.LongitudeDestino
);
        String[] kmETempo = distanciaCalculada.Split(';');
        novaDistancia.TotalKm = Convert.ToDouble(kmETempo[0]);
        novaDistancia.TempoEmSegundo = Convert.ToInt64(kmETempo[1]);
        novasDistancias.Add(chave, distanciaCalculada);
    }
    catch (Exception ex)
    {throw new Exception(ex.Message);}
    distanciaRetorno = novaDistancia;
    distanciasCalculadas.Add(chave, novaDistancia);
}
return distanciaRetorno;

```

3.3.2.5 Rotina de roteirização do Bing

No quadro 8 pode-se visualizar a rotina que implementa o serviço de roteirização do *web service* do Bing. Esta rotina recebe como parâmetro a origem e destino através de coordenadas geográficas. O Bing retorna a distância em quilômetros e o tempo em minutos do trajeto, essas informações são obtidas de trajeto sempre utilizando modal rodoviário.

Quadro 8 – Rotina de roteirização consumida da *WebService* da API do Bing

```

public string calcularDistanciaETempo
( double latitudeOrigem, double longitudeOrigem,
  double latitudeDestino, double longitudeDestino)
{
    string key = "suachave";
    BingRoutingService.RouteRequest request = new RouteRequest();
    request.Credentials = new
Roteirizador.Servico.BingRoutingService.Credentials { ApplicationId =
key };

    request.Waypoints = new Waypoint[]
    {
        new Waypoint(){ Location = new
Roteirizador.Servico.BingRoutingService.Location
        Latitude = latitudeOrigem,
        Longitude= longitudeOrigem}},
        new Waypoint()
        { Location = new
Roteirizador.Servico.BingRoutingService.Location()
        { Latitude = latitudeDestino,
        Longitude= longitudeDestino
        }
        },
    };

    request.UserProfile = new
Roteirizador.Servico.BingRoutingService.UserProfile
    {
        DistanceUnit =
Roteirizador.Servico.BingRoutingService.DistanceUnit.Kilometer
    };

    request.Options = new RouteOptions()
    {
        Mode = TravelMode.Driving,
        Optimization = RouteOptimization.MinimizeTime
    };

    RouteServiceClient client = new RouteServiceClient();
    RouteResponse response = client.CalculateRoute(request);
    RouteResult result = response.Result;
    return result.Summary.Distance + ";" + result.Summary.TimeInSeconds;
}

```

3.3.2.6 Rotina de geração de um mapa no Bing Maps

No quadro 9 pode-se visualizar a rotina que recebe um JSON de todas as coordenadas geográficas na ordem que o sistema calculou e salva em um *array*. Após é chamado o método `createDrivingRoute` que desenha os pontos e traça a rota entre eles conforme as informações do *array*. No quadro 10 pode-se visualizar a rotina que gera o mapa.

Quadro 9 – Rotina que recebe os pontos geográficos

```

Var position = { "coordenadas": [] };
loadMap();
$.ajax({
  url: '@Url.Action("MontarMapa","Roteirizar")',
  type: 'POST',
  async: false,
  data: { 'id': idRoteirizador, 'numeroRoteiro': $("#Rota").val() },
  success: function (data) {
    $.each(data, function (i, coordenadasGeograficas) {
      var coords = { "lat": "", "lon": "" };
      coords.lat = coordenadasGeograficas.Latitude;
      coords.lon = coordenadasGeograficas.Longitude;
      position.coordenadas.push(coords);
    });
  }
});
if (!directionsManager) {
  Microsoft.Maps.loadModule('Microsoft.Maps.Directions', { callback:
createDrivingRoute });
}

```

Quadro 10 – Rotina que gera o mapa

```

for (var i = 0; i < position.coordenadas.length; i++)
{
  var trace = position.coordenadas[i];
  startWaypoint = new Microsoft.Maps.Directions.Waypoint({
location: new Microsoft.Maps.Location(trace.lat, trace.lon) });
  if (i == 0) {
    primeiro = startWaypoint;
  }
  directionsManager.addWaypoint(startWaypoint);
}
directionsManager.addWaypoint(primeiro);
directionsManager.setRequestOptions({ distanceUnit:
Microsoft.Maps.Directions.DistanceUnit.kilometers });

directionsManager.setRenderOptions({ itineraryContainer:
document.getElementById('itineraryDiv') });

Microsoft.Maps.Events.addHandler(directionsManager,
'directionsError', displayError);

directionsManager.calculateDirections();

```

3.3.2.7 Chamada da rotina nova geração

No quadro 11 pode-se visualizar o trecho de código que executa a rotina nova geração. Dependendo da configuração de hardware do computador que está executando o algoritmo, esta rotina pode criar várias *Threads* para executá-las.

Quadro 11 – nova geração de cromossomos

```

long processadores = Environment.ProcessorCount;
Thread[] Threads = new Thread[processadores];
for (int i = 0; i < processadores; i++)
{
    Threads[i] = new Thread(() => novaGeracao.Execute());
    Threads[i].Name = "threadFilha " + i;
}

for (int i = 0; i < processadores; i++)
{
    Threads[i].Start();
}

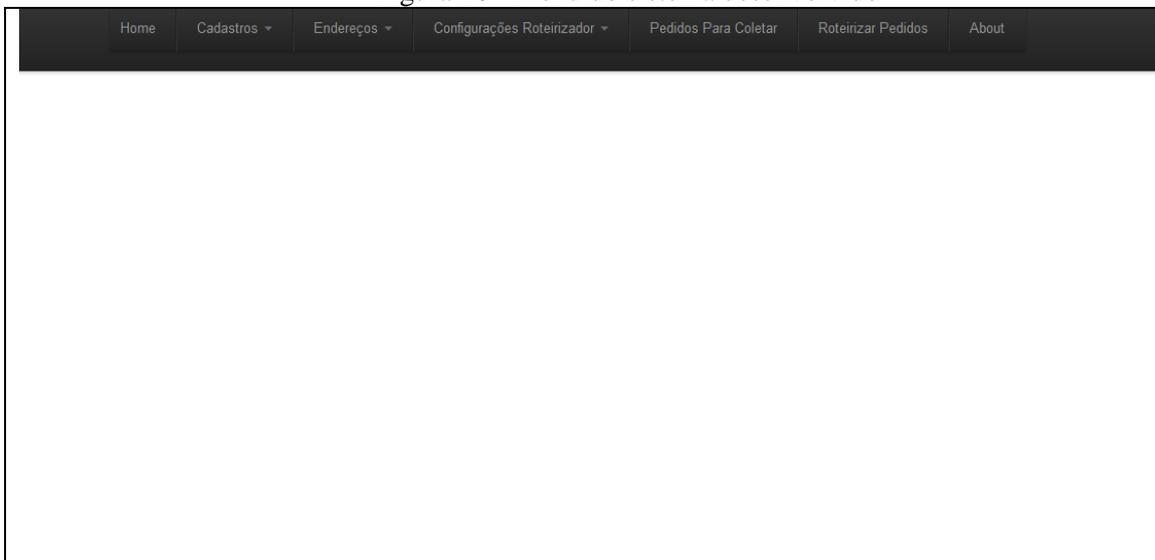
for (int i = 0; i < processadores; i++)
{
    Threads[i].Join();
}

```

3.3.3 Operacionalidade da implementação

A seguir são destacadas as principais funcionalidades presentes no sistema. Na figura 16 é mostrado o menu da ferramenta.

Figura 16 - Menu do sistema desenvolvido



A estrutura das telas do sistema é formada por um topo, contendo o menu principal e na parte central são exibidas as informações referentes ao item selecionado no menu. Cada opção possui uma listagem dos registros com opções de excluir e editar registro (figura 17). Com exceção as telas de Roteirizar pedidos e o menu de configurações de roteirizador que possuem um comportamento diferente.

Figura 17 - Tela de depósito

Nome	Cidade	Logradouro	Cep	Latitude	Longitude
Transporte Osika	Blumenau	Belo Horizonte, 487	89025-001	-26,9671478	-49,0641899
Rua Mariana Zabel	Blumenau	Rua Mariana Zabel	89066-220	-26,8353043	-49,1021423

3.3.3.1 Cadastros

Através do menu superior (figura 18) é possível acessar as informações de cadastros. Os cadastros estão subdivididos em 4 itens principais para facilitar navegação da ferramenta, que são:

- cadastros básicos: no item de cadastro básico é possível realizar o cadastro ou manutenção de clientes, depósitos, produtos e veículos (figura 19);
- cadastro endereço: através desse menu é possível efetuar o cadastro ou manutenção de cidades, estado e País (figura 20);
- configurações roteirizador: neste menu é possível realizar configurações do algoritmo genético, restrições fixas e cadastrar novas restrições (figura 21);
- pedidos para coletar: através deste menu, é possível realizar o cadastro ou manutenção de pedidos para coletas.

Figura 18 – Menu principal da ferramenta

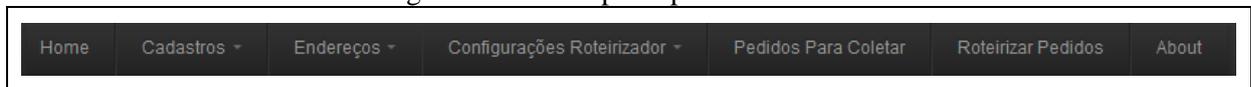


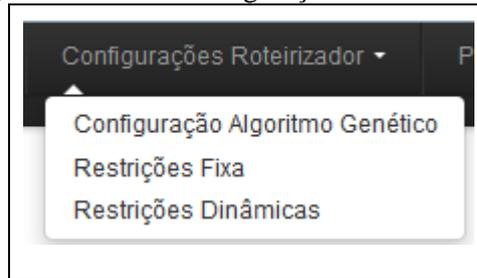
Figura 19 – Menu cadastros e subitens



Figura 20 – Menu cadastros de endereços e subitens

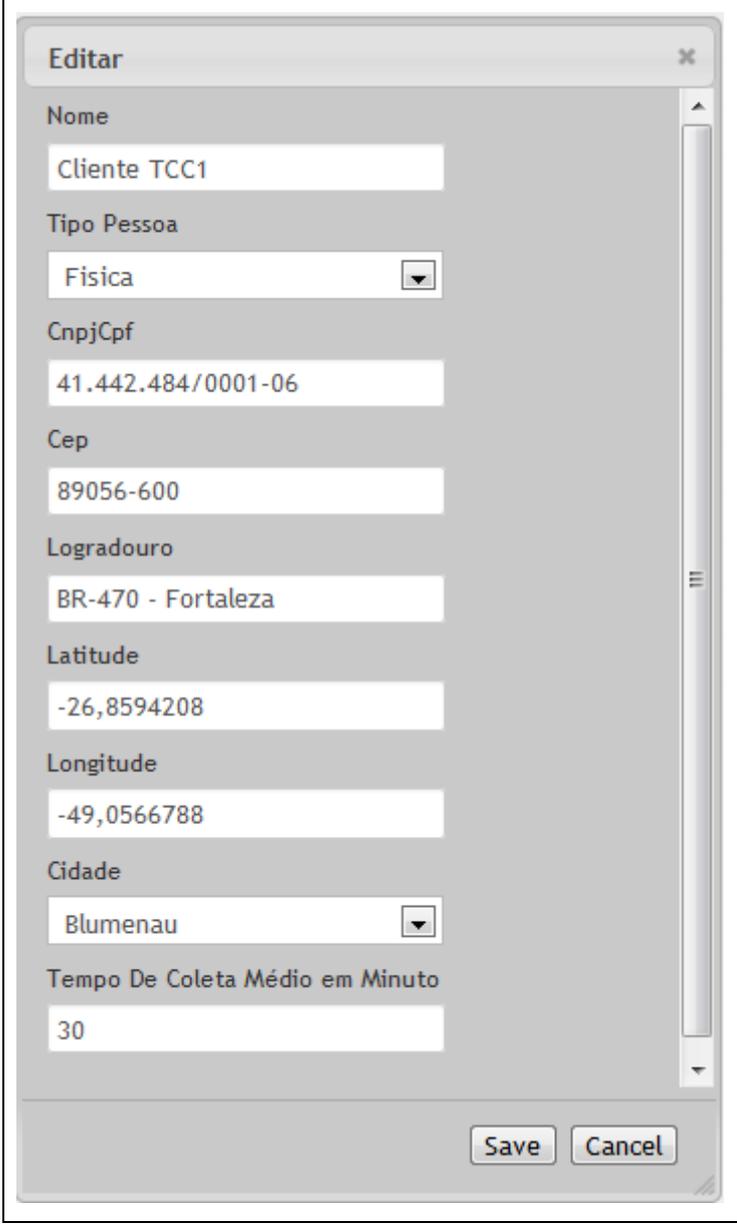


Figura 21 – Menu configurações roteirizador



Conforme a figura 22 no cadastro de clientes são informada o Cnpj e seu endereço. Os atributos latitude e longitude são opcionais. Caso não seja preenchido, o processo de roteirização quando processar pedidos do cliente irá fazer comunicação com o *web service* do Bing e irá retornar as informações de latitude e longitude e atualizar no cadastro.

Figura 22 – Cadastro de cliente



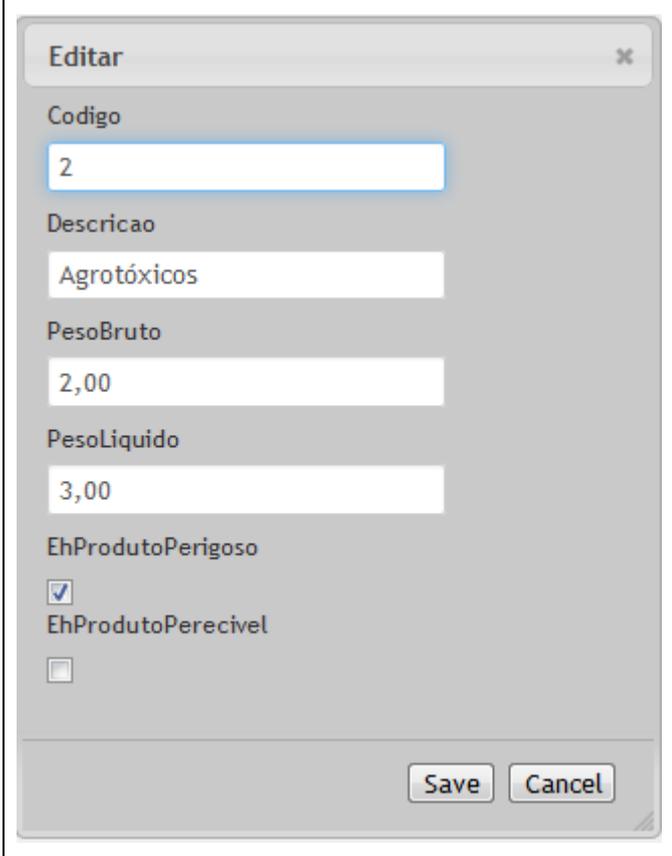
The image shows a software window titled "Editar" (Edit) with a close button (X) in the top right corner. The window contains a form for editing a client record. The fields are as follows:

Field Name	Value
Nome	Cliente TCC1
Tipo Pessoa	Fisica
CnpjCpf	41.442.484/0001-06
Cep	89056-600
Logradouro	BR-470 - Fortaleza
Latitude	-26,8594208
Longitude	-49,0566788
Cidade	Blumenau
Tempo De Coleta Médio em Minuto	30

At the bottom right of the form, there are two buttons: "Save" and "Cancel".

Na figura 23 demonstra o cadastramento de um produto no sistema.

Figura 23 – Cadastro de produtos



The image shows a software window titled "Editar" with a close button (X) in the top right corner. The window contains the following fields and controls:

- Codigo:** A text input field containing the number "2".
- Descricao:** A text input field containing the text "Agrotóxicos".
- PesoBruto:** A text input field containing the value "2,00".
- PesoLiquido:** A text input field containing the value "3,00".
- EhProdutoPerigoso:** A checkbox that is checked (indicated by a blue checkmark).
- EhProdutoPercível:** A checkbox that is unchecked.

At the bottom right of the window, there are two buttons: "Save" and "Cancel".

No cadastro de veículo são informados os campos: placa, peso, largura, altura e tipo de veículo. O campo capacidade em metros cúbicos é usado na restrição de capacidade de veículo. A figura 24 mostra o cadastramento do veículo CLU3003.

Figura 24 – Cadastro de veículos

Editar

Descricao
CLO3013

Placa
CLU3013

Peso
100000,00

Largura
1000000,00

Altura
20000,00

CapacidadeMc
70000,00

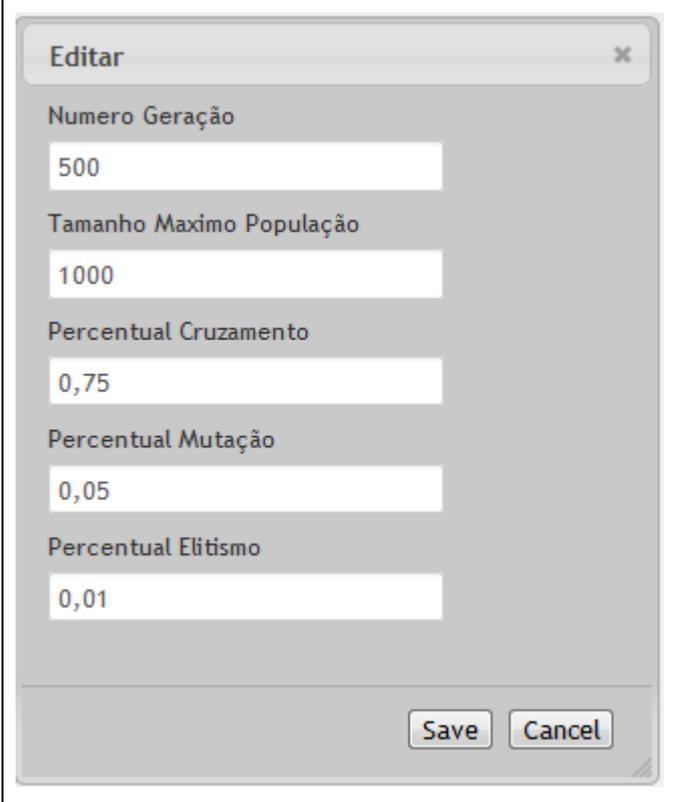
CustoPorKM
10,00

TipoVeiculo
Caminhao Leve

Save Cancel

Na figura 25 mostra a tela de configuração do algoritmo genético. Podem ser observado que é possível alterar o número de geração, tamanho da população, percentual de cruzamento, percentual de mutação e taxa de elitismo.

Figura 25 – Configurações do algoritmo genético

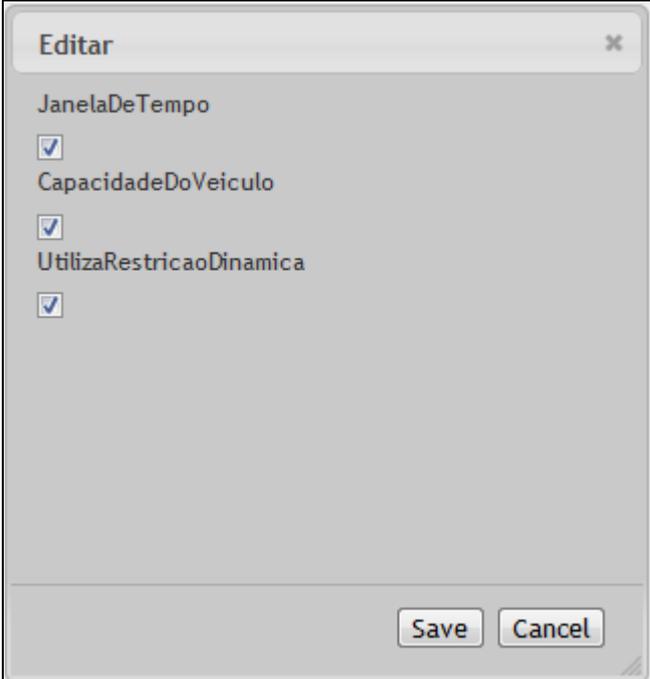


The image shows a dialog box titled "Editar" with a close button (X) in the top right corner. It contains five input fields for configuration parameters, each with a label above it and a text box below it. At the bottom right, there are two buttons: "Save" and "Cancel".

Parameter	Value
Numero Geração	500
Tamanho Maximo População	1000
Percentual Cruzamento	0,75
Percentual Mutação	0,05
Percentual Elitismo	0,01

A Figura 26 demonstra uma tela de parametrização com as restrições do sistema. Podem ser observado que é possível habilitar ou desabilitar as restrições de Janela de Tempo, capacidade do veículo ou as restrições dinâmicas.

Figura 26 – Parametrizar restrição fixas



The image shows a dialog box titled "Editar" with a close button (X) in the top right corner. It contains four checkboxes, each with a label above it. At the bottom right, there are two buttons: "Save" and "Cancel".

Constraint	Status
JanelaDeTempo	<input checked="" type="checkbox"/>
CapacidadeDoVeiculo	<input checked="" type="checkbox"/>
UtilizaRestricaoDinamica	<input checked="" type="checkbox"/>

O sistema possibilita montar restrições dinâmicas, baseadas nas entidades do modelo de negócio do sistema e de alguns atributos customizados implementados. Na Figura 27, pode-se visualizar as restrições cadastradas.

Figura 27 – Lista de restrição dinâmicas

Restrição	Tipo Filtro	Condição	Tipo		
1	Produto	Igual	EhPerigoso		⊖
1	Veiculo	Igual	TipoVeiculo	UtilitarioLeve	⊖
2	TempoPercurso	Igual	TempoMaximoPercursoEmHora	8	⊖

Pode-se observar que existem 3 itens cadastrados. Contudo, dois destes registros possuem na coluna restrição o valor 1 e outro item o valor 2. A rotina de roteirização agrupa todas as restrições por número da restrição, transformando em uma restrição com várias condições. Portanto, existem duas restrições que pode-se ser visualizado no quadro 12 as suas anotações.

Quadro 12 – Restrições dinâmicas cadastradas

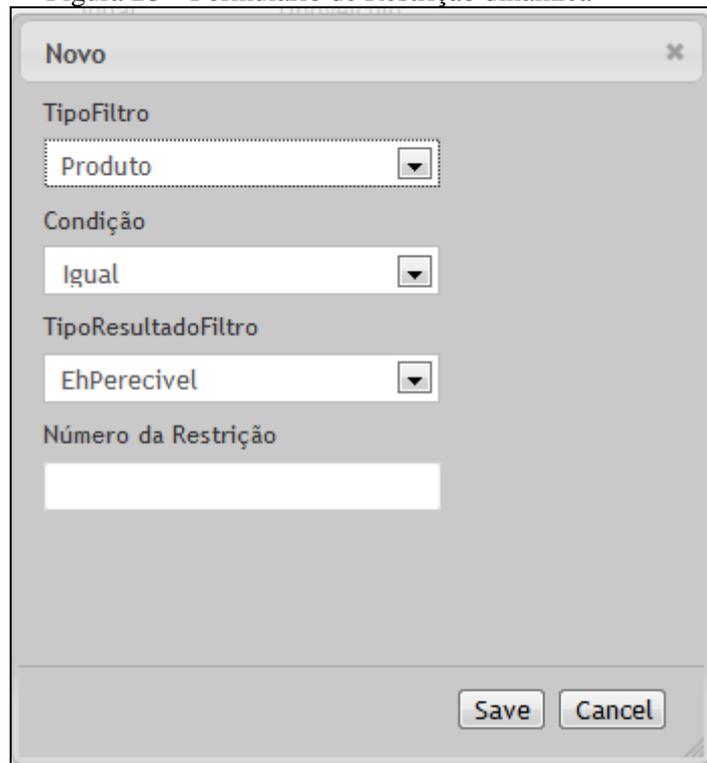
Restrição 1: Se Produto Igual Tipo EhPerigoso E Se Veiculo Igual TipoVeiculo Igual UtilizarioLeve

Restrição 2: Se TempoPercurso Igual TempoMaximoPercursoEmHora Igual 8 Horas

Para inserir uma restrição é necessário selecionar um filtro, conforme a seleção é carregado no *comboBox* `TipoResultadoFiltro` os campos disponíveis para serem selecionados. Conforme o tipo primitivo do `TipoResultadoFiltro` selecionado pode aparecer um novo campo para selecionar um valor ou para ser digitado um número ou até um texto. Exemplo, caso seja selecionado um campo do tipo inteiro o sistema irá carregar um campo para ser digitado um valor. Nesta, mesma tela é necessário selecionar uma condição que pode ser igual, maior que e menor que.

Na Figura 28 pode-se visualizar a tela de inserção de restrição.

Figura 28 – Formulário de Restrição dinâmica



Novo

TipoFiltro
Produto

Condição
Igual

TipoResultadoFiltro
EhPercivel

Número da Restrição

Save Cancel

The image shows a software dialog box titled 'Novo' (New) for creating a dynamic restriction. It contains four dropdown menus: 'TipoFiltro' (Filter Type) set to 'Produto', 'Condição' (Condition) set to 'Igual', and 'TipoResultadoFiltro' (Filter Result Type) set to 'EhPercivel'. Below these is an empty text input field for 'Número da Restrição' (Restriction Number). At the bottom right are 'Save' and 'Cancel' buttons.

No Quadro 13 é possível visualizar as restrições dinâmicas disponíveis no sistema, cada filtro selecionado possui determinados atributos para montar as condições conforme o seu modelo de classe desenvolvido. Foi observada uma importância de ter implementado o filtro TempoPercurso, devido a uma lei da legislação Brasileira, que não permite um motorista dirigir mais que 4 horas de forma interrupto.

Quadro 13 – Listagem de atributos disponíveis para montar restrições

Filtro	Tipo Resultado Filtro
Produto	Percivel EhPerigoso PesoBruto PesoLiquido Descricao Código
Veiculo	Código TipoVeiculo CapacidadeMc Placa Chassi Renavam Ativo CustoPorKM
Cliente	Cidade TempoDeColetaMedioEmMinuto Longitude Latitude Cep CnpjCpf TipoPessoa
PedidoColeta	PesoCubado Numero Cliente Deposito DataInicio DataFim
TempoPercurso	TempoMaximoPercuroEmHora

A tela de pedidos de coletas contém a aba geral e os itens de pedidos. Aba geral pode ser visualizada na Figura 29. E a aba itens dos pedidos pode ser visualizado na Figura 30.

Figura 29 – Formulário de Pedido de Coleta

Editar

Geral **Itens Pedido**

Número
19

Início Do Primeiro Período
15/03/2014 08:00:29

Fim Do Primeiro Período
15/03/2014 12:30:29

Início Do Segundo Período
15/03/2014 14:00:00

Fim Do Segundo Período
15/03/2014 18:50:29

Deposito
Transporte Osika

Cliente
Cliente TCC1

Peso Cubado
60000,00

Save

Figura 30 - Itens do Pedido

Editar

Itens Pedido

Numero Nota :
1200

Peso Total :
100

Quantidade Volume :
12

Total Nota :
500,00

Produto:
Agrotóxicos

NumeroNota	PesoTotal	QuantidadeVolume	TotalNota	Produto	Itens
No data available in table					

3.3.3.2 Roteirizar Pedidos

Através do menu (figura 18) é possível acessar a tela de roteirização de coletas. Na Figura 31 pode-se visualizar a tela.

Figura 31 - Roteirização

Coletas De: 24/05/2014 Até: 24/05/2014 Hora de Saída: 08:00

Origem: Transporte Osika

Rota	Veiculo Utilizado	Origem	Destino	Custo Veiculo KM	Km Percorrido	Tempo Percorrido
Total KM Total Custo Veiculo KM Tempo do Percurso						
0	0					00:00:00

Na tela de roteirização é obrigatório confirmar algumas informações para iniciar o processo, como as coletas que serão roteirizadas, o depósito que é responsável pelas mesmas e o horário de saída dos veículos. As coletas que serão roteirizadas são filtradas através do intervalo de data e o campo origem.

Após a execução da roteirização é mostrado o resultado conforme a figura 32.

Figura 32 – Resultado da roteirização

The screenshot shows a web interface for route optimization. At the top, there are input fields for 'Coletas De: 15/03/2014', 'Até: 15/03/2014', and 'Hora de Saída: 08:00'. Below these is a dropdown menu for 'Origem: Transporte Osika'. A prominent 'Roteirizar' button is visible. The main part of the interface is a table with the following data:

Rota	Veículo Utilizado	Origem	Destino	Custo Veículo KM	Km Percorrido	Tempo Percorrido
1	CLU3505	Transporte Osika	Cliente TCC2	141,825	9,455	00:21:30
1	CLU3505	Cliente TCC2	Transporte Osika	129,705	8,647	00:16:33
2	CLU3013	Transporte Osika	Cliente TCC1	154,09	15,409	00:24:55
2	CLU3013	Cliente TCC1	Cliente TCC3	256,49	25,649	00:34:00
2	CLU3013	Cliente TCC3	Transporte Osika	387,39	38,739	00:56:03

Below the table, there are summary statistics and two buttons: 'Visualizar Mapa' and 'Visualizar Trace da Solução'.

Total KM	Total Custo Veículo KM	Tempo do Percurso
97,899	1069,5	02:33:01

Na figura 32 pode-se visualizar que a roteirização gerou uma tabela. Nela contém informações do número do roteiro gerado, veículo que será utilizado, origem, destino, o custo do veículo por quilômetro rodado, a quantidade de quilômetro percorrido e o tempo. E nesta roteirização gerou duas rotas uma para o veículo da placa é CLU3505 e outra para a placa CLU3013.

A apresentação visual das rotas calculadas é feita através do uso do Bing Maps API. O mapa das figuras 33 e 34 retratam as rotas geradas através das informações demonstradas na figura 32.

Quadro 14 - Comparação com trabalhos correlatos

Funcionalidade	Osika (2014)	Miura (2003)	Rosa (2005)	Tsuda (2007)
Plataforma	Web	Desktop	Desktop	Desktop
Disponibiliza integração da rotina de roteirização com outro sistemas	Sim	Não	Não	Não
Recursos gráficos para visualizar a rota.	Sim	Não	Não	Não
Janelas de Tempo	Sim	Sim	Não	Não
Restrição de capacidade do veículo	Sim	Sim	Sim	Sim
Restrição dinâmica	Sim	Não	Não	Não
Possui mecanismo de geocodificação	Sim	Não	Não	Não
Método	Algoritmo Genético	Clarke e Wright	CSP	<i>Sweep algorithm</i>

Para confrontar a qualidade da solução foram realizados dois experimentos comparando-os com a solução do trabalho de Tsuda (2007) que comparou com um cenário de teste real. Conforme o quadro 14 pode-se observar que o trabalho de Tsuda (2007), utiliza restrições para verificar a capacidade do veículo. Portanto, para comparar esta solução com a solução proposta neste trabalho, foi necessário desabilitar as restrições de janela de tempo e restrição dinâmica.

Para elaborar o cenário de teste Tsuda (2007) extraiu informações de um dia de operação de uma determinada empresa estudada e confrontou o percurso realizado pela empresa sem auxílio de ferramentas de otimizações de rota com o percurso realizado através da sua solução.

A entrada dos dados deste problema encontra-se no Anexo A e B deste trabalho.

Quadro 15 – Comparação dos resultados obtidos com 15 pontos de coletas

Solução	Rota dos Veículos	Distância Total
Osika	Rota do veículo 1 0- 8-15-6-5-11-7-2-1-12-14-4-0 Rota do veículo 2 0-3-13-10-9-0	89 km
Tsuda	Rota do veículo 1 0-3-5-11-6-15-8-9-10-0 Rota do veículo 2 0-4-14-12-2-7-1-13-3-0	88,8 km
Situação Real	Rota do veículo 1 0-3-0 -4-3-10-8-9-0 Rota do veículo 2 0-13-1-14-12-2-7-11-5-6-15-0	109,6 km

Pode-se visualizar os parâmetros configurados do algoritmo genético no quadro 16.

Quadro 16 – Parâmetros obtidos dos resultados com 15 coletas

Tamanho da população: 4000
Taxa de cruzamento: 0.90
Taxa de mutação: 0.50
Número da geração: 500
Taxa de elitismo: 0.1

A capacidade do veículo utilizado neste cenário foi de 60 metros cúbicos para o veículo 1 e 85 para o 2.

A entrada dos dados deste problema encontra-se nos Anexos C e D deste trabalho.

Quadro 17 - Comparação dos resultados obtidos com 24 pontos de coletas

Solução	Rota dos Veículos	Distância Total
Osika	Rota do veículo 1 0 -13 - 17 - 15 - 2 - 1 - 3 - 11 - 14 - 22 - 16 - 24 - 0 = 53.3 Rota do veículo 2 0 - 5 - 12 - 6 - 4 - 7 - 9 - 10 - 8 - 19 - 18 -23 - 21 - 20 - 0 = 82.5	135.83 km
Tsuda	Rota do veículo 1 0 - 24 - 16 - 22 - 14 - 11 - 3 -1 - 2 - 15 - 17 - 13 - 0 = 53,3414 Rota do veículo 2 0 - 5 - 12 - 6 - 4 - 7 - 9 - 10 - 8 - 19 - 18 - 23 - 21 - 20 - 0 = 82,4824	135.8239 km
Situação Real	Rota do veículo 1 0 - 5 - 6 - 7 - 3 - 1 - 2 - 0 = 93,0224 Rota do veículo 2 0 - 12 - 18 - 23 -19 - 9 - 10 - 8 - 11 - 14 - 22 - 16 - 13 - 17 - 15 - 24 - 21 - 20 - 0 = 82,7518	175.7742 km

Pode-se visualizar os parâmetros configurados do algoritmo genético no quadro 18.

Quadro 18 – Parâmetros do roteiro com 24 pontos de coletas

Tamanho da população: 5000
Taxa de cruzamento: 0.90
Taxa de mutação: 0.10
Número da geração: 500
Taxa de elitismo: 0.1

A capacidade do veículo utilizado neste cenário foi de 60 metros cúbicos para ambos os veículos.

Podemos observar nos quadros 15 e 17 que a distância total percorrida pela rotina implementada neste trabalho é praticamente a mesma que a do trabalho de Tsuda, contudo os roteiros divididos foram e gerados de forma diferente.

O algoritmo genético implementado se mostra eficaz para encontrar uma boa solução utilizando as restrições propostas. Contudo, a qualidade da solução e o tempo de execução do algoritmo variam conforme alguns fatores.

- a) número de restrição do problema: após diversos testes foi observado que quanto maior for o número de restrições do problema, maior deverá ser os parâmetros do algoritmo genético, com a finalidade de tentar explorar todo o espaço de busca e encontrar uma boa solução;
- b) tamanho da população: o tamanho , de busca impacta diretamente no tempo de execução do algoritmo. Com uma população grande foi observado que o espaço de busca é maior. Contudo, o tempo de execução do algoritmo aumenta muito;
- c) número de geração: este parâmetro possui a responsabilidade de executar o algoritmo várias vezes com a finalidade de evoluir a população. Com um número alto foi observado que aumenta o tempo de execução do algoritmo, contudo há probabilidade de explorar o espaço de busca é maior.

No quadro 19, são apresentados alguns testes de execução do algoritmo com a finalidade de avaliar a qualidade da solução e o tempo de processamento. As linhas em negritos são soluções ótimas.

Quadro 19 – Testes na execução do AG

Gerações	Tamanho da população	Quantidade de Thread	Melhor Solução	Tempo
500	5000	4	135.83 km	15: 25
500	4000	4	137.75 km	11: 96
500	2000	4	138.63 km	05: 04
500	1000	4	169.45 km	02: 49
500	2000	1	153.3 km	14:09
500	1000	1	157.61 km	07:50
1000	5000	4	135.83 km	35:94
1000	4000	4	138.63 km	27: 20
1000	2000	4	149.27 km	12: 72
1000	1000	4	162.07 km	02: 14

Com a série de testes apresentada no Quadro 19, podem se notar algumas características como: o tamanho da população e o número de geração interfere na qualidade da solução e o tempo de execução do algoritmo. Já executando com mais Threads melhora o tempo de execução do algoritmo.

Para os testes foram utilizados o experimento do quadro 17. A máquina utilizada nos testes foi um Intel Core i3, 2.53 GHz, 4 megabytes de memória e sistema operacional Windows 7.

4 CONCLUSÕES

De forma em geral, podemos dizer que o objetivo deste trabalho foi alcançado. Os resultados obtidos nos testes mostrou que o sistema roteiriza soluções de forma eficiente, sempre respeitando as restrições impostas no problema. O que ajuda a minimizar custo e tempo da complexa operação de transporte de coleta, já que são gerados os roteiros a ser percorridos por determinados veículos.

Este trabalho apresentou um algoritmo genético para resolução do problema que mostrou-se flexível às restrições impostas pelo problema e totalmente extensível para novas restrições. Os operadores genéticos conseguiram evoluir a população nos experimentos realizados até conseguir ao menos uma solução boa.

Em relação ao tempo de processamento da solução, foi implementada a estratégia de dividir para conquistar para que o número de gerações do algoritmo funcione em paralelo em mais de uma *Thread*. Desta forma foi possível em uma velocidade maior obter a resolução do problema.

O sistema foi desenvolvido na plataforma C# (ASP.NET) e banco de dados SQL Server. Foram utilizados padrões e tecnologias atuais, como ASP.NET MVC, *Entity Framework*, WEB API. O sistema permitiu visualizar as rotas geradas da funcionalidade de roteirização em um mapa através da API do Bing Maps. A API do Bing Maps mostrou-se importante e precisa para retornar informações geográficas para o sistema. Outra funcionalidade importante foi ter criado um *Web Service* utilizando a tecnologia Web API que permite integração da função de roteirização com outros sistemas de logística.

Destacam-se como limitação problemas na evolução da população quando o espaço de busca é muito grande observou-se que os operadores genéticos de cruzamento e mutação, podem precisar de outra estratégia para explorar determinados espaços de busca e continuar evoluindo os cromossomos.

Por fim, este trabalho serve como base para que outros trabalhos possam ser desenvolvidos academicamente.

4.1 EXTENSÕES

Como sugestões para continuação deste trabalho e melhoria do sistema, pode-se citar:

- a) implementar estratégias de busca local para que em conjunto com o algoritmo genético melhore a sua evolução;
- b) otimizar o processamento do algoritmo genético;

- c) implementar alguma estratégia para que os parâmetros do algoritmo genético sejam obtidos de forma dinâmica conforme o número de variáveis do problema a ser roteirizado;
- d) desenvolver um aplicativo para dispositivo móvel para que seja possível visualizar o roteiro gerado em um mapa;
- e) criar a entidade empresa, onde ela poderá ter mais de um depósito;
- f) implementar operação de entrega e gerar a roteirização com entregas e coletas simultâneas.

REFERÊNCIAS

- ALMEIDA, Mayron R. de et al. **Algoritmos genéticos aplicados a um caso real do problema de roteamento de veículos**. Fortaleza, 2007. Disponível em: <<http://www.din.uem.br/sbpo/sbpo2007/pdf/arq0010.pdf>>. Acesso em: 20 ago. 2013.
- ARAÚJO, Carlos E. G. **Algoritmos genéticos híbridos sem delimitadores de rotas para problemas de roteirização de veículos**. 2008. 89 f. Dissertação (Mestrado em Engenharia de Sistemas Logísticos) – Escola Politécnica da Universidade de São Paulo, São Paulo. Disponível em: <http://www.teses.usp.br/teses/disponiveis/3/3148/tde-28032008-161354/publico/Dissertacao_Carlos_Eduardo_v13.pdf>. Acesso em: 20 ago. 2013.
- ASSIS, Luciane P. de. **Algoritmo para o problema de roteamento de veículo para coleta e entrega simultâneas**. 2007. 67 f. Dissertação (Pós-Graduação em Ciência da Computação) - Universidade Federal de Minas Gerais, Minas Gerais. Disponível em: <<http://dcc.ufmg.br/pos/cursos/defesas/873M.PDF>>. Acesso em: 20 ago. 2013.
- BALLOU, Ronald H. **Gerenciamento da cadeia de suprimentos: planejamento, organização e logística empresarial**. Porto Alegre: Bookman, 2001.
- BARAT, J. **Logística e transporte no processo de globalização oportunidades para o brasil**. 1. ed. São Paulo: Unesp, 2007
- BOECK, Kelly C. **Aplicação web para consulta de itinerários de transporte público com visualização no google maps**. 2012. 81 f. Trabalho de Conclusão de Curso (Bacharelado em Ciências da Computação) – Universidade Regional de Blumenau, Blumenau. Disponível em: <http://www.bc.furb.br/docs/MO/2012/350378_1_1.pdf>. Acesso em: 20 dez. 2013.
- CUNHA, Claudio B. **Aspectos práticos da aplicação de modelos de roteirização de veículos a problemas reais**. *Transportes*, v.8, n.2, p.51-74, nov. 2000. Disponível em: <http://sites.poli.usp.br/ptr/ptr/docentes/cbcunha/files/roteirizacao_aspectos_praticos_CBC.pdf>. Acesso em: 17 ago. 2013.
- DIAS, Marco A. P. **Administração de materiais: uma abordagem logística**. São Paulo: Atlas, 2010.
- GOMES, Carlos F. S.; RIBEIRO, Priscilla C. C. **Gestão da cadeia de suprimentos integrada à tecnologia da informação**. 1. ed. São Paulo: Pioneira Thomsom Learning, 2004.
- GUIA DO TRANSPORTADOR. **Guia do transportador**, 2014. Disponível em: <http://www.guiadotrc.com.br/logistica/visao_sistematica.asp>. Acesso em: 03 mar. 2014.
- LINDEN, RICARDO. **Algoritmo genético: uma importante ferramenta da inteligência computacional**. 2. ed. Rio de Janeiro: Brasport Livros e Multimídia, 2008.

LOBO, Eduardo L. M. **Uma solução do problema de horário escolar via algoritmo genético paralelo**. 2005. 80 f. Dissertação (Mestrado em Modelagem Matemática e Computacional) – Centro Federal de Educação Tecnológica de Minas Gerais, Belo Horizonte. Disponível em: <<http://www.mmc.cefetmg.br/info/downloads/D006-EduardoLuizMirandaLobo2005.pdf>>. Acesso em: 03 mar 2014.

LUCAS, Diogo C. **Algoritmos genéticos: uma introdução**. Porto Alegre, 2002. Disponível em: <<http://www.inf.ufrgs.br/~alvares/INF01048IA/ApostilaAlgoritmosGeneticos.pdf>>. Acesso em: 03 mar. 2014.

MALAGUIAS, Neli. G. L. **Uso dos algoritmos genéticos para a otimização de rotas de distribuição**. 2006. 113 f. Dissertação (Pós-Graduação Em Engenharia Elétrica) - Universidade Federal de Uberlândia, Minas Gerais. Disponível em: <<http://repositorio.ufu.br/bitstream/123456789/358/1/UsoAlgoritmosGen%C3%A9ticos.pdf>>. Acesso em: 20 ago. 2013.

MIURA, M. **Resolução de um problema de roteamento de veículos em uma empresa transportadora**. 2003. 127 f. Monografia (Mestrado em Engenharia da Produção) – Escola Politécnica da Universidade de São Paulo, São Paulo. Disponível em: <<http://alvarestech.com/temp/vrptw/vrp-tranporte.pdf>>. Acesso em: 20 ago. 2013.

MOURA, B. **Logística: conceitos e tendências**. 1. ed. Lisboa: Centro Atlântico, 2000.

PIRES, Alexsandro S. **Implementação de um algoritmo heurístico para problemas de restrições**. 2006. 58 f. Trabalho de Conclusão de Curso (Bacharelado em Ciências da Computação) – Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau. Disponível em: <<http://campeche.inf.furb.br/tccs/2006-II/2006-2alexsandropiresvf.pdf>>. Acesso em: 20 ago. 2013.

PRESTES, Álvaro N. **Uma análise experimental de abordagens heurísticas aplicadas ao problema do caixeiro viajante**. 2006. 74 f. Monografia (Mestrado em Sistema e Computação) – Universidade Federal do Rio Grande do Norte, Rio Grande do Norte.

RIBEIRO, Jorge C. P. **Logística de estoque**. 2001. 47 f. Monografia (Pós-Graduação Em Lato Sensu em Marketing no Mercado Globalizado), Universidade Cândido Mendes, Rio de Janeiro. Disponível em: <<http://www.avm.edu.br/monopdf/24/JORGE%20CLAUDIO%20PLASTINA%20RIBEIRO.pdf>>. Acesso em: 03 mar. 2014.

RODRIGUES, Paulo R. A. **Introdução aos sistemas de transporte do Brasil e à logística internacional**. São Paulo: Yone Silva Pontes, 2007.

ROSA, Viviane B. **Sistema para logística de distribuição**. 2005. 67 f. Trabalho de Conclusão de Curso (Bacharelado em Ciências da Computação) – Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau. Disponível em: <http://www.bc.furb.br/docs/MO/2005/306772_1_1.pdf>. Acesso em: 20 ago. 2013.

RUSSEL, Stuart J.; NORVIG, Peter. **Inteligência artificial**. Rio de Janeiro: Elsevier, 2004.

SENS, Thomas A. **Sistema de pedidos de pizza para otimização de rotas no google maps.** 2009. 56 f. Trabalho de Conclusão de Curso (Bacharelado em Ciências da Computação) – Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau. Disponível em: <<http://campeche.inf.furb.br/tccs/2009-II/TCC2009-2-25-VF-ThomasASens.pdf>>. Acesso em: 20 ago. 2013.

SUCUPIRA, Igor R. **Programação por propagação de restrições: teoria e aplicações.** São Paulo, 2003. Disponível em: <<http://www.ime.usp.br/~igorrs/ic/relatorio/relatorio.pdf>>. Acesso em: 20 ago. 2013.

TSUDA, Diogo S. **Modelo de roteirização de veículos em uma empresa importadora de produtos japoneses.** 2007. 144 f. Trabalho de Conclusão de Curso (Graduação em Engenharia de Produção) – Escola Politécnica da Universidade de São Paulo, São Paulo. Disponível em: <<http://pro.poli.usp.br/wp-content/uploads/2012/pubs/modelo-de-roterizacao-de-veiculos-em-uma-empresa-importadora-de-produtos-japoneses.pdf>>. Acesso em: 20 ago. 2013.

ANEXO A – Informações dos clientes do estudo de caso 1

Na figura 35 pode-se visualizar as informações da demanda e coordenadas geográficas dos clientes usados no estudo de caso 1.

Figura 35 Clientes do estudo de caso 1

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

Fonte: Tsuda (2007, p. 70).

ANEXO B – Matriz de distância do estudo de caso 1

Na figura 36 pode-se visualizar a Matriz de distância de todos os clientes utilizados no estudo de caso 1.

Figura 36 – Matriz de distância do estudo de caso 1

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

Fonte: Tsuda (2007, p. 71).

ANEXO C – Informações dos clientes do estudo de caso 2

Na figura 37 pode-se visualizar as informações da demanda e coordenadas geográficas dos clientes usados no estudo de caso 2.

Figura 37 - Clientes do estudo de caso 2

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

Fonte: Tsuda (2007, p. 123).

ANEXO D – Matriz de distância do estudo de caso 2

Na figura 38 pode-se visualizar a Matriz de distância de todos os clientes utilizados no estudo de caso 2.

Figura 38 - Matriz de distância do estudo de caso 2

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

Fonte: Tsuda (2007, p. 124).