

FACULDADE DE TECNOLOGIA DE SÃO JOSÉ DOS CAMPOS

PAULO RENATO MORAIS OLIVEIRA

**UM ALGORITMO GENÉTICO PARA O PROBLEMA ROTEAMENTO DE
VEÍCULOS**

SÃO JOSÉ DOS CAMPOS

2011

PAULO RENATO MORAIS OLIVEIRA

**UM ALGORITMO GENÉTICO PARA O PROBLEMA DE ROTEAMENTO DE
VEÍCULOS**

TRABALHO DE GRADUAÇÃO
APRESENTADO À FACULDADE DE
TECNOLOGIA DE SÃO JOSÉ DOS
CAMPOS, COMO PARTE DOS
REQUISITOS NECESSÁRIOS PARA A
OBTENÇÃO DO TÍTULO DE
TECNÓLOGO EM BANCO DE DADOS.

Orientador: Reinaldo GenIchiro Arakaki

Co-orientador: Ubirajara Moura de Freitas

SÃO JOSÉ DOS CAMPOS

2011

PAULO RENATO MORAIS OLIVEIRA

MÓDULO PARA ROTEAMENTO DE VEÍCULOS

TRABALHO DE GRADUAÇÃO
APRESENTADO À FACULDADE DE
TECNOLOGIA DE SÃO JOSÉ DOS
CAMPOS, COMO PARTE DOS
REQUISITOS NECESSÁRIOS PARA A
OBTENÇÃO DO TÍTULO DE
TECNÓLOGO EM BANCO DE DADOS.

ANTÔNIO EGYDIO SÃO THIAGO GRAÇA

ANDERSON VIÇOSO DE ARAÚJO

REINALDO GEN ICHIRO ARAKAKI

UBIRAJARA MOURA DE FREITAS

___/___/___

DATA DE APROVAÇÃO

A persistência é o que leva a perfeição

Marcelo D2

AGRADECIMENTOS

Agradeço aos meus familiares e amigos por todo o suporte e compreensão nas horas mais difíceis.

Aos professores da Fatec pelo incentivo e o conhecimento compartilhado.

Agradeço a todos aqueles que me acompanharam durante essa jornada de três anos e me ajudaram a vencer mais esta etapa da vida.

RESUMO

Para as empresas se manterem competitivas e conquistarem um mercado cada vez mais concorrido tornou-se fundamental a necessidade aperfeiçoar os processos de distribuição, dentre todos os processos envolvidos na cadeia logística o transporte é aquele que absorve a maior parcela do custo, reduzir esse custo significa diminuir o preço final do produto, além de aumentar a lucratividade.

Devido à importância do transporte viário na economia mundial a otimização das rotas tornou-se muito importante para reduzir custos e o tempo de distribuição. A necessidade de otimizar rotas para realizar as entregas é conhecida como problema de roteamento de veículos ou PRV.

Para implementação deste trabalho foi desenvolvido um protótipo para simular o roteamento de veículos para um depósito e uma lista de clientes, no qual os clientes estão espalhados em um mapa e cada um possui sua respectiva coordenada e quantidade de produtos solicitada. A frota de veículos é considerada homogênea, ou seja, todos possuem veículos possuem mesma capacidade.

O objetivo do trabalho é criar um algoritmo genético para roteamento de veículos para otimizar as rotas de distribuição no qual sejam respeitadas as restrições de capacidade dos veículos e que a distância total percorrida por eles sejam mínimas.

ABSTRACT

For companies remain competitive and gain an increasingly competitive market, it became fundamental the need to improve distribution processes, among all the processes involved in logistics chain, transport is one that absorbs the largest share of the cost, reduce the cost means lowering the price and increase profitability.

Given the importance of road transport in the world economy, optimize routes became very important to reduce distribution costs and time. This need is known as vehicle routing problem or VRP.

To implement this work tests, a prototype was developed to simulate the vehicles routing to a deposit and a client list, in which these customers are spread across a map and each has its corresponding coordinate and quantity of products ordered. the vehicle fleet is considered homogeneous, in other words, all vehicles have the same capacity.

The purpose of this work is to create a genetic algorithm for routing of vehicles to optimize delivery routes in which the restrictions of vehicle capacity and the total distance covered by them are minimal.

LISTA DE FIGURAS

Figura 1 – Dígrafo ponderado com o caminho mínimo.....	13
Figura 2 - Grafo orientado ou dígrafo.....	19
Figura 3 - Grafo não orientado	19
Figura 4 - Grafo orientado ponderado	20
Figura 5- No momento da reprogramação das rotas, o problema pode ser visto como um PRVJT.....	22
Figura 6 - Exemplo do funcionamento do algoritmo de Dijkstra.....	27
Figura 7- Descrição dos passos do algoritmo genético.....	30
Figura 8 - Gráfico de probabilidade de seleção com o método de seleção proporcional	35
Figura 9 - Seleção pelo método de torneio.....	37
Figura 10 - Exemplo da seção de mapeamento do pmx (MALAQUIAS, 2006)	39
Figura 11 -Cópia dos elementos da seção de mapeamento dos pais para os filhos.....	39
Figura 12 - Cópia dos elementos restantes no pai e a troca pelos elementos da seção de mapeamento correspondente	40
Figura 13 - Operador cruzamento de um único ponto	41
Figura 14 - Exemplo de indivíduos para o operador de cruzamento de ordem.....	42
Figura 15 - Cópia dos elementos dos pais para os filhos no operador de cruzamento de ordem	42
Figura 16 - Resultado da execução do operador de cruzamento de ordem.....	43
Figura 17- Mutação por troca	44
Figura 18 - Mutação por inversão simples	44
Figura 19 - Mutação por deslocamento.....	45
Figura 20 - Mutação por inserção.....	46
Figura 21- Mutação por inversão.....	47
Figura 22 - Mutação scramble.....	47
Figura 23 - Exemplo de solução para um problema com nove pontos de entrega.....	49
Figura 24 - Exemplo de cromossomo	49
Figura 25 - Rota extraída de um cromossomo.....	50
Figura 26 - Cromossomo adaptado para não repetir o centro de distribuição.....	50
Figura 27- Diagrama de classes da solução proposta para o PRV clássico	52
Figura 28 - Tela de execução do protótipo.....	53
Figura 29 - Gráfico comparativo entre os métodos de seleção	57
Figura 30- Gráfico comparativo entre os operadores de mutação	58
Figura 31 - Gráfico comparativo entre os operadores de cruzamento	58
Figura 32 - Gráfico comparativo de diferentes tamanho de população	59
Figura 33 - Gráfico comparativo entre diferentes taxas de mutação.....	60
Figura 34 - Gráfico comparativo entre diferentes taxas de cruzamento	61

LISTA DE TABELAS

Tabela 1 - Distância entre os clientes	32
Tabela 2 - Coordenadas do depósito.....	55
Tabela 3 - Coordenadas dos clientes.....	55
Tabela 4 - Parâmetros utilizados como padrão.....	56
Tabela 5 - Comparação entre os resultados obtidos com as melhores soluções conhecidas para várias instâncias do problema de roteirização de veículos	62

SUMÁRIO

1	INTRODUÇÃO	12
1.1	Motivação.....	12
1.2	Objetivos	14
1.2.1	Objetivo Geral	14
1.2.2	Objetivos Específicos	14
1.3	Metodologia	15
1.4	Organização do Trabalho	15
2	PROBLEMA DE ROTEAMENTO DE VEÍCULOS	17
2.1	Definição do problema.....	17
2.2	Grafos.....	18
2.3	Variações do PRV	20
2.3.1	Problema de Roteamento de Veículos com Janela de Tempo	21
2.3.2	Problema de Roteamento de Veículo com Janela de Tempo Dinâmico.....	21
2.3.3	Problema de Roteamento de Veículos com Pedidos de Coleta e Entrega Simultâneo	23
2.4	Problema de Roteamento de Veículos com Pedidos de Coleta e Entrega ...	24
2.5	Complexidade do PRV	24
2.6	Problema do caminho mínimo	25
2.6.1	Algoritmo de Dijkstra	25
2.6.2	Algoritmo de Bellman-Ford.....	27
2.6.3	Algoritmo A Estrela (Algoritmo A*)	28
3	ALGORITMO GENÉTICO (AG)	29
3.1	Representação das soluções viáveis.....	30
3.1.1	Representação Binária	31
3.1.2	Representação por inteiros	31
3.2	Função de custo	32
3.3	Função de aptidão.....	33
3.4	População inicial	33
3.5	Seleção	34
3.5.1	Dizimação	34
3.5.2	Seleção proporcional.....	35

3.5.3	Torneio	36
3.6	Operadores genéticos	37
3.6.1	Cruzamento (<i>Crossover</i>)	38
3.6.2	Cruzamento de mapeamento parcial (PMX)	38
3.6.3	Cruzamento de ponto único	40
3.6.4	Cruzamento de ordem (OX)	41
3.7	Mutação	43
3.7.1	Mutação por troca (EM)	43
3.7.2	Mutação por inversão simples (SIM)	44
3.7.3	Mutação por deslocamento (DM)	45
3.7.4	Mutação por inserção (ISM)	45
3.7.5	Mutação por inversão (IVM)	46
3.7.6	Scramble Mutation (SM)	47
4	PROPOSTA DE SOLUÇÃO	48
4.1	Representação de uma solução	48
4.2	Modelagem da solução proposta para a resolução do PRV clássico	51
5	RESULTADOS COMPUTACIONAIS	54
5.1	Problema teste	54
5.2	Parâmetros do algoritmo genético	56
5.3	Comparação com resultados encontrados na literatura	61
6	CONSIDERAÇÕES FINAIS	63
6.1	Contribuições e conclusões	63
6.2	Trabalhos futuros	64
	REFERÊNCIAS BIBLIOGRÁFICAS	66

1 INTRODUÇÃO

1.1 Motivação

O conforto e a praticidade que o transporte rodoviário proporciona ao seu usuário contribuíram para que houvesse um crescimento significativo na frota de veículos do país, provocando um aumento no número de engarrafamentos, acidentes e desgaste das vias (LEMOS, 2009).

Para empresas transportadoras que atuam no segmento rodoviário esses fatores aliados ao custo de manutenção da sua frota de veículos encarecem o valor dos serviços prestados, pesquisas indicam que os custos relacionados à distribuição representam de 10% a 15% do valor final das mercadorias (ALVARENGA, 2005), tornando-se necessário buscar formas de aperfeiçoar seus processos para reduzir custos e criar diferenciais sobre seus concorrentes (ALVARENGA, 2005).

O problema de roteamento de veículos (PRV) tem sido estudado com muito interesse devido à necessidade de diminuição dos gastos na distribuição dos produtos (BERRI, 2004), isto pode ser realizado com a utilização de um sistema de informação geográfica (SIG) que determine a rota com o menor custo, este custo pode ser determinado através do cálculo total da distância percorrida ou o tempo gasto com o trajeto.

Existem algumas variações do PRV como, por exemplo, o problema do caixeiro viajante (PCV) que consiste em determinar uma rota que minimize a distancia total percorrida e que assegure que o caixeiro viajante visite todas as cidades exatamente uma vez (DAVIS, 1997).

Pode ser considerada fácil de determinar melhor solução, basta calcular o custo de todas as combinações possíveis e selecionar a que apresenta o menor custo, esta estratégia de testar todas as soluções possíveis é conhecida como força bruta. Isto é viável quando o tamanho do conjunto de dados é pequeno, à medida que o conjunto

crece esta solução se torna inviável do ponto de vista computacional, isto ocorre, pois o número de combinações cresce de forma exponencial - $O(c^n)$, no qual c é uma constante e n é a quantidade de vértices do grafo (HEINEN, 2005).

Em termos de complexidade computacional o PRV é classificado como sendo NP-Difícil assim como a maioria dos problemas combinatoriais. Desta forma, para solucioná-los é necessário a utilização de métodos heurísticos e meta-heurísticos (HEINEN, 2005).

Uma forma comum de representar o PRV é através de um grafo, no qual se mostra muito eficiente para representar uma malha rodoviária, onde os vértices podem representar os cruzamentos, as arestas as vias e o peso das arestas o custo do trajeto (tempo de trajeto ou a distância percorrida). Por se tratar de roteamento de veículos serão utilizados grafos orientados ponderados, também conhecidos como dígrafos ponderados, pois com eles é possível indicar a direção do fluxo e o custo do trajeto. Sendo assim, para determinar a rota de menor custo, é necessário determinar o vértice de origem e o vértice de destino e então calcular qual o caminho com menor peso entre os vértices conforme mostrado na figura 1.

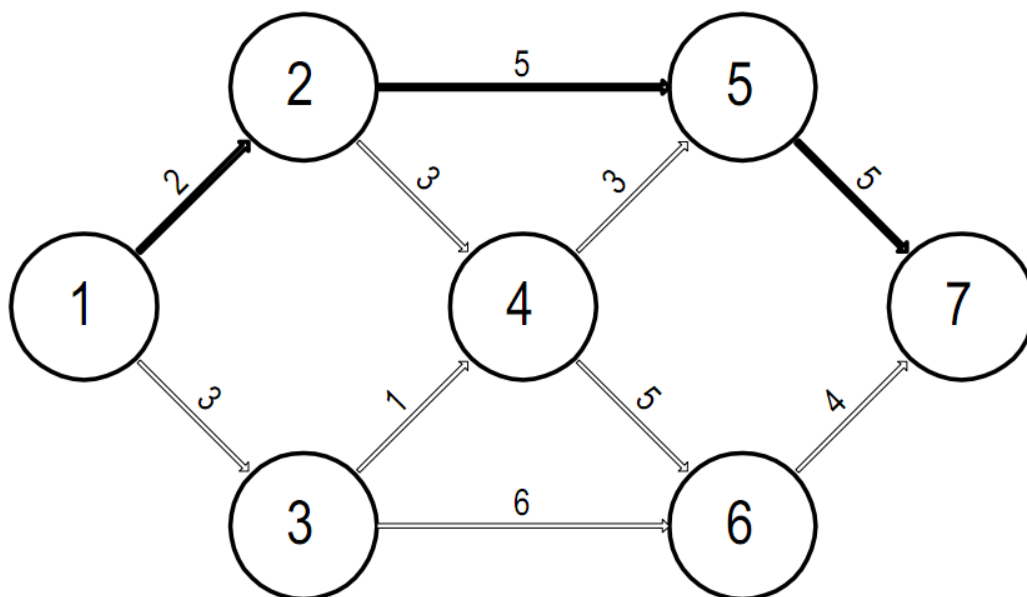


FIGURA 1 – DÍGRAFO PONDERADO COM O CAMINHO MÍNIMO

1.2 Objetivos

1.2.1 Objetivo Geral

O objetivo deste trabalho é criar um algoritmo genético para roteamento de veículos para otimizar as rotas de distribuição no qual sejam respeitadas as restrições de capacidade dos veículos e que a distância total percorrida por eles sejam mínimas.

1.2.2 Objetivos Específicos

Os objetivos específicos deste trabalho são:

- a. Pesquisar técnicas e algoritmos da área de otimização combinatória para resolução do PRV;
- b. Desenvolver um algoritmo genético para o PRV para cálculo de rotas de distribuição respeitando a capacidade dos veículos e minimizando o custo total;
- c. Pesquisar e implementar operadores genéticos para uma renovação eficaz da população afim de melhorar as soluções obtidas;
- d. Realizar testes capazes de provar a qualidade das soluções obtidas comparadas com soluções conhecidas de algumas instâncias do PRV;
- e. Criar uma representação gráfica para a apresentação dos resultados obtidos e simulação das rotas,

1.3 Metodologia

Para a solução do problema proposto foi implementado um protótipo para solução do problema de roteamento de veículos clássico, no qual a frota de veículos é homogênea. Foi utilizado um algoritmo genético com no qual o usuário pode modificar os parâmetros, selecionar o método de seleção e os operadores genéticos de mutação e cruzamento desejados.

A linguagem de programação utilizada foi Java, pois é uma linguagem orientada a objetos, gratuita e multi-plataforma, ou seja, funciona para várias arquiteturas distintas, sistemas operacionais distintos e trabalha com paradigmas de programação distintos como desktop e web.

1.4 Organização do Trabalho

Este trabalho está organizado da seguinte forma:

- a) Capítulo 2: Este capítulo tem como objetivo introduzir ao conceito de problema de roteamento de veículos e suas variações, as técnicas utilizadas para resolução de problemas de roteamento de veículos.
- b) Capítulo 3: Este capítulo apresenta o conceito de algoritmo genético, formas de representar uma solução, os passos de um algoritmo genético, métodos de seleção e os operadores genéticos de mutação e cruzamento.
- c) Capítulo 4: Este capítulo apresenta uma proposta de solução utilizando um algoritmo genético para o problema de roteamento de veículos com restrição de capacidade dos veículos.
- d) Capítulo 5: Este capítulo apresenta o problema utilizado para teste, uma comparação entre os resultados obtidos variando os parâmetros passados ao

algoritmo genético e uma comparação dos resultados obtidos para algumas instâncias do problema.

- e) Capítulo 6: Este capítulo apresenta os resultados e conclusões obtidos com o desenvolvimento deste trabalho bem como as propostas de trabalhos futuros

2 PROBLEMA DE ROTEAMENTO DE VEÍCULOS

Este capítulo tem como objetivo introduzir ao conceito de problema de roteamento de veículos e suas variações e as técnicas utilizadas para resolução de problemas de roteamento de veículos.

2.1 Definição do problema

A otimização das rotas tornou-se muito importante para reduzir custos e o tempo de distribuição devido à importância do transporte viário na economia mundial. Esta necessidade de otimizar as rotas para realizar as entregas é conhecida como problema de roteamento de veículos ou PRV

O PRV clássico (com restrições de capacidade e distância máxima) consiste em designar um conjunto de rotas de entrega e/ou coleta tal que (SOSA, 2007):

- O custo total do conjunto de rotas percorrido pela frota seja minimizado;
- Cada rota se inicie e finalize no depósito;
- Cada cliente tenha sua demanda suprida exatamente por um veículo;
- A carga total de cada veículo não exceda sua capacidade;
- O tempo total necessário para completar qualquer rota não exceda um limite pré-especificado, que inclui tempos de viagem entre clientes e tempos de serviço em cada cliente.

2.2 Grafos

Um grafo G consiste de dois conjuntos V e E . V é um conjunto finito e não vazio de vértices e E é um conjunto de pares não ordenados de vértices chamados arestas. $V(G)$ e $E(G)$ representam os conjuntos de vértices e arestas do grafo G , também denotado por $G=(V,E)$ (CORMEN, 2002).

Um grafo orientado G , também conhecido por dígrafo, é um par (V, E) , onde V é um conjunto finito e E é uma relação binária em V . O conjunto V é conjunto de vértices de G e seus elementos são chamados vértices. O conjunto E é chamado conjunto de arestas de G , e seus elementos são chamados arestas, a figura 2 mostra um exemplo deste tipo de grafo (CORMEN, 2002).

Em um grafo não-orientado $G = (V, E)$, o conjunto de arestas E consiste em pares de vértices não ordenados, em lugar de pares ordenados. Por convenção usa-se a notação (u, v) para uma aresta, a figura 3 mostra um exemplo deste tipo de grafo.

Muitas definições para grafos orientados e não orientados são idênticas, embora alguns termos tenham um significado diferente nos dois contextos. Se (u, v) é uma aresta em um grafo orientado $G = (V, E)$, é dito que (u, v) é incidente do ou sai do vértice u e é incidente no ou entra no vértice v . Por exemplo, as arestas que saem do vértice b da figura 2 são (b, e) e (b, f) . As arestas que entram no vértice b são (a, b) e (c, b) . Se (u, v) é uma aresta em grafo não-orientado $G = (V, E)$, é dito que (u, v) é incidente nos vértices u e v . Por exemplo, na figura 4 as arestas incidentes no vértice b são (a, b) , (e, b) e (c, e) (CORMEN, 2002).

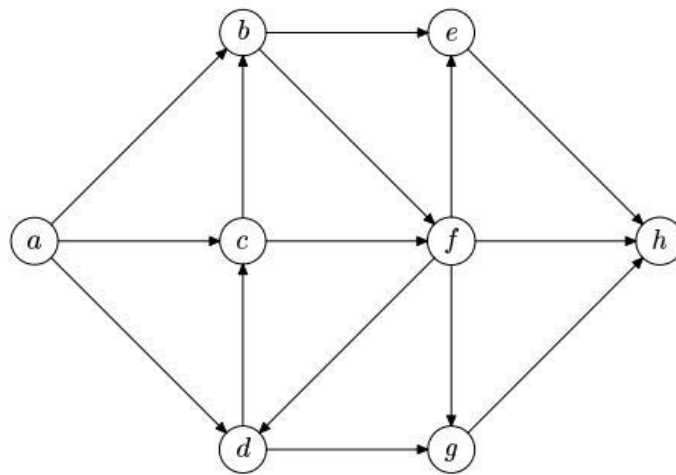


FIGURA 2 - GRAFO ORIENTADO OU DÍGRAFO

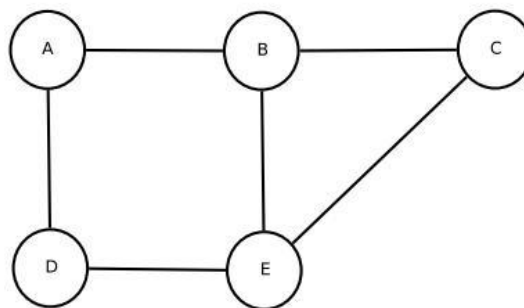


FIGURA 3 - GRAFO NÃO ORIENTADO (ROSA, 2008)

Por se tratar de um problema de roteamento de veículos serão utilizados grafos orientados ponderados também conhecidos como dígrafos ponderados, pois assim é possível indicar a direção do fluxo e o custo do trajeto, conforme exemplificado na figura 4. Portanto para determinar o caminho de menor custo, é necessário determinar o vértice de origem e o vértice de destino, então calcular qual o caminho com menor peso entre os vértices.

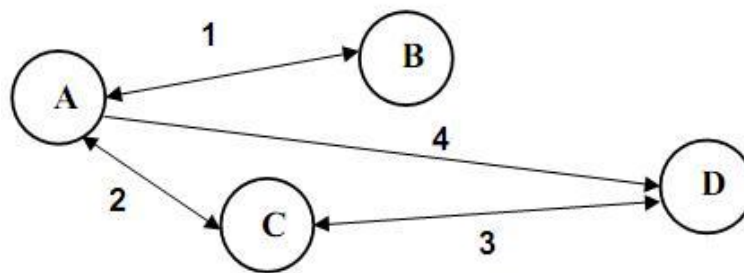


FIGURA 4 - GRAFO ORIENTADO PONDERADO

2.3 Variações do PRV

Podem existir inúmeras variações do PRV, cada uma se adapta a um conjunto de problemas diferentes, conforme pode ser encontrado o problema de roteamento de veículos estático e dinâmico com janela de tempo em (ALVARENGA, 2005), (DALBONI, 2003), o problema de roteamento de veículos em (MALAQUIAS, 2006), o problema de roteamento de veículos com pedidos de coleta e entrega em (MONTANÉ,2006) entre outros.

O PRV e suas variações têm sido objeto de estudo de muitos pesquisadores, devido a sua complexidade, procurando otimizar o desempenho dos algoritmos é proposto a utilização de diversas técnicas de otimização combinatória como Algoritmos Genéticos (MALAQUIAS, 2006) e (ALVARENGA, 2005), Simulated Anneling (GALVÃO, 2007), Busca Tabu (MONTANÉ, 2006) entre outras.

Nas seções seguintes serão apresentadas algumas variações do PRV.

2.3.1 Problema de Roteamento de Veículos com Janela de Tempo

O problema de roteamento de veículos com janela de tempo (PRVJT) consiste em planejar rotas para uma frota de veículos, que além da limitação da capacidade, inclui a restrição da janela de tempo, ou seja, o veículo tem que chegar a um determinado cliente dentro de um intervalo de tempo (ALVARENGA, 2005).

Esse problema pode ser definido sobre um grafo $G = (V, A)$ sendo $V = \{v_0, v_1, v_2, \dots, v_n\}$ um conjunto de vértices e $A = \{(v_i, v_j) : v_i, v_j \in V, i \neq j\}$ é o conjunto de arestas. O vértice v_0 representa o depósito e os outros vértices os clientes que devem ser atendidos (GRAÇA, 2009).

Dada uma frota homogênea de m veículos, o objetivo do PRVJT é encontrar um conjunto de rotas que inicie e termine no depósito representado pelo vértice v_0 . De tal modo que as seguintes regras sejam atendidas (GRAÇA, 2009):

- Cada veículo deve atender uma única rota;
- Cada vértice a partir de v_1 deve ser visitado somente uma vez;
- Os veículos devem partir do depósito somente após a abertura do depósito e retornar antes do fechamento do depósito;
- No caso de o veículo chegar ao cliente em um tempo menor que a janela de tempo, este veículo deverá aguardar, caracterizando-se uma espera calculada.

2.3.2 Problema de Roteamento de Veículo com Janela de Tempo Dinâmico

No Problema de Roteamento de Veículo com Janela de Tempo Dinâmico (PRVJTD) alguns clientes não são conhecidos no momento em que os veículo saem

para fazer as entregas, à medida que as novas requisições são inseridas tem-se um novo PRVJT estático para ser resolvido (GRAÇA, 2009).

A figura 5 exemplifica a transformação do PRVJTD em PRVJT. Essa figura mostra a situação de duas rotas no momento em que uma nova requisição é realizada, sendo que os círculos preenchidos correspondem às posições atuais de cada veículo. Nesse caso, existe um conjunto de rotas existentes no qual cada uma inicia na posição atual do veículo e termina no depósito (RIBEIRO, 2005).

Observa-se que para inserir a nova requisição, assim como no problema estático, o objetivo é minimizar o custo, logo um PRVJTD pode ser resolvido considerando várias instâncias de PRVJT estáticos (GRAÇA, 2009).

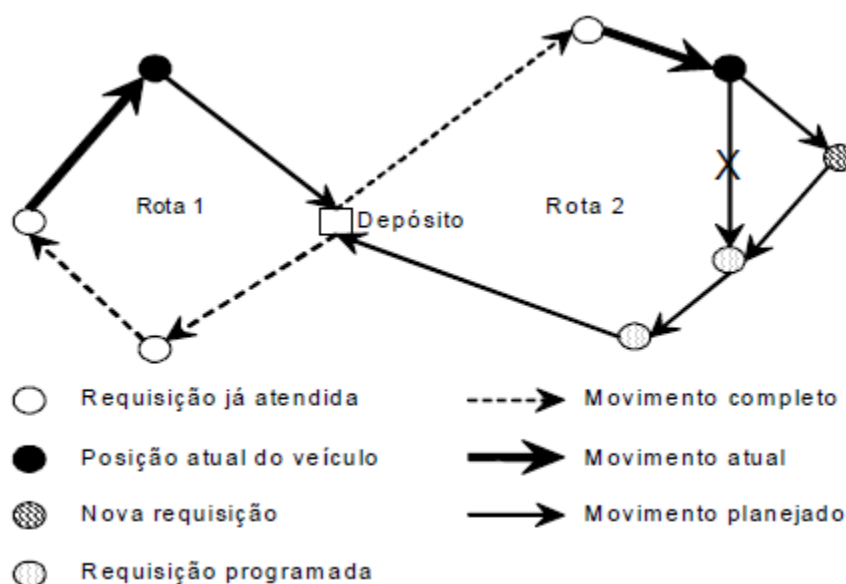


FIGURA 5- NO MOMENTO DA REPROGRAMAÇÃO DAS ROTAS, O PROBLEMA PODE SER VISTO COMO UM PRVJT (RIBEIRO, 2005)

Para sejam respeitadas as restrições de janela de tempo é necessário que as novas requisições sejam realizadas antes de um tempo limite para que possam ser atendidas no mesmo dia, as requisições que chegarem após o tempo limite serão agendadas para o próximo dia. Como não existe conhecimento prévio da localização das novas requisições, uma central de comunicação informa ao motorista as mudanças em sua rota (GRAÇA, 2009).

2.3.3 Problema de Roteamento de Veículos com Pedidos de Coleta e Entrega Simultâneo

No Problema de Roteamento de Veículos com Pedidos de Coleta e Entrega Simultâneos (PRVPES) os clientes fazem pedidos de coleta e entrega, os quais devem ser atendidos de forma simultânea pelo mesmo veículo. Uma característica deste tipo de problema é que a carga do veículo é composta por dois itens de naturezas distintas, este tipo de problema é visualizado com frequência na indústria de alimentos e bebidas, na qual além de entregar os novos produtos o veículo deve recolher os produtos com validade vencida (MONTANÉ, 2006).

Esse problema pode ser definido sobre um grafo $G = (V, A)$ sendo $V = \{v_0, v_1, v_2, \dots, v_n\}$ um conjunto de vértices e $A = \{(v_i, v_j) : v_i, v_j \in V, i \neq j\}$ é o conjunto de arestas. O vértice v_0 representa o depósito que serve de base para as entregas (MONTANÉ, 2006).

Cada vértice $v_i \in V$ representa um cliente que tem associada duas demandas: coleta e entrega. Considerando-se que o número de veículos disponíveis é ilimitado. O PRVCES consiste em determinar as rotas que serão utilizadas de maneira que (MONTANÉ, 2006):

- As rotas se iniciam no depósito e terminam no depósito;
- Cada cliente é visitado exatamente uma vez;
- Ambas as demandas dos clientes são atendidas;
- A carga máxima na rota não excede a capacidade do veículo utilizado;
- A distância total percorrida é minimizada.

2.4 Problema de Roteamento de Veículos com Pedidos de Coleta e Entrega

No Problema de Roteamento de Veículos com Pedidos de Coleta e Entrega (PRVPE) o cliente pode fazer somente um tipo de pedido (coleta ou entrega). Em ambos os casos, existem dois tipos de carga no veículo, a carga que ainda não foi entregue e a carga que já foi recolhida (MONTANÉ, 2006).

Para definir o PRVCE será usada a mesma definição do PRVCES, pois este é um caso particular do PRVCES no qual uma das demandas é zero (MONTANÉ, 2006).

2.5 Complexidade do PRV

O PRV é um problema NP-difícil, devido à sua complexidade computacional, métodos de solução exata são inviáveis para instâncias de grande porte, fazendo-se necessário a utilização de métodos heurísticos ou meta-heurísticos. Como pode ser encontrado em (PIMENTA, 2001), (ALVARENGA, 2005), (MALAQUIAS, 2006), (SOSA, 2007), (GRAÇA, 2009) entre outros (SOSA, 2007).

Sua complexidade de tempo é exponencial - $O(c^n)$, onde c é uma constante e n é o número de nós a serem analisados. Até o momento, não foi possível encontrar nenhuma solução de tempo polinomial para problemas da classe NP-Completo. (HEINEN, 2005).

2.6 Problema do caminho mínimo

O problema do caminho mínimo consiste em minimizar o custo de travessia entre dois nós de um grafo, custo este dado pela soma dos pesos de cada aresta percorrida.

Os algoritmos especializados em solucionar o problema do caminho mínimo são chamados de algoritmos de busca de caminhos. Nas seções seguintes serão apresentados os algoritmos mais conhecidos.

Para resolver um problema de caminho mínimo a rede viária é modelada em forma de grafo orientado ponderado, os vértices representam os entroncamentos, as arestas representam as vias e o peso das arestas que representam o custo do trajeto. (PIMENTA, 2001)

O custo de uma rota pode levar em conta diversos fatores, esses fatores dependem do propósito do sistema e por quem este será utilizado, por exemplo, para transporte de uma carga valiosa, a empresa pode levar em consideração o índice de criminalidade da região.

Outros fatores que podem ser levados em consideração são a distância a ser percorrida, velocidade máxima de cada trecho, a intensidade do tráfego, o número de acidentes registrados, entre outros (BERRI, 2004).

2.6.1 Algoritmo de Dijkstra

O algoritmo de Dijkstra soluciona o problema do caminho mínimo em um grafo orientado com suas arestas de peso não negativas, porém podendo aceitar valores nulos,

esta restrição é perfeitamente possível no contexto de redes de transportes, onde as arestas representam normalmente distâncias ou tempos médios de percurso (CORMEN, 2002).

O algoritmo considera um grafo G , composto por um conjunto de vértices e um conjunto de arestas, os vértices são divididos em três grupos: os já visitados, os candidatos ou adjacentes e os nunca visitados ou desconhecidos. A figura 6 exemplifica o funcionamento do algoritmo de Dijkstra (DAVIS, 1997).

A priori são conhecidos dois nós do grafo chamados de origem e destino, o conjunto de visitados é inicializado com o vértice origem e a distância do vértice origem para todos os nós é definida como infinita, o conjunto de candidatos é inicializado com os vizinhos imediatos do vértice origem (DAVIS, 1997).

A cada passo do algoritmo, os nós do conjunto candidatos são verificados para determinar qual seria a melhor opção para expandir a pesquisa, é calculado a distância do nó atual para os seus vizinhos e se esta distância for menor que a gravada como menor caminho para aquele vértice que inicialmente foi marcada como infinita e o valor é atualizado. (DAVIS, 1997).

Após escolhido o melhor candidato este será transferido para o conjunto de visitados e seus adjacentes para o conjunto de candidatos (DAVIS, 1997).

O algoritmo para quando o vértice destino for alcançado ou quando o não há mais vértices para serem visitados, ou seja, todos os vértices já foram visitados e não foi encontrado nenhum caminho viável entre origem e destino (DAVIS, 1997).

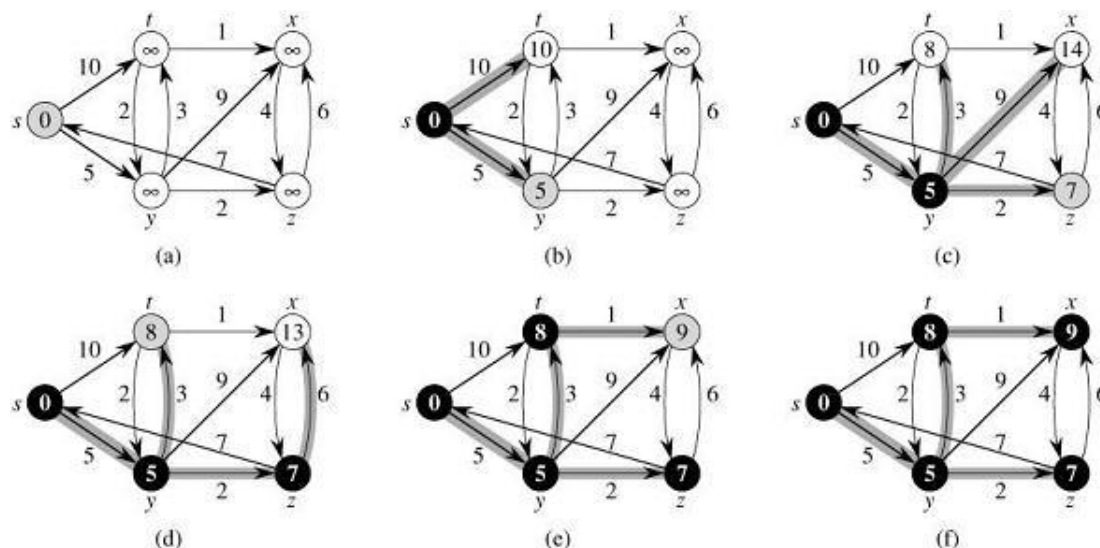


FIGURA 6 - EXEMPLO DO FUNCIONAMENTO DO ALGORITMO DE DIJKSTRA (CORMEN, 2002)

Este algoritmo calcula o custo mínimo do vértice origem para todos os demais vértices do grafo. O algoritmo pode ser usado sobre grafos orientados ou não (DAVIS, 1997).

2.6.2 Algoritmo de Bellman-Ford

Resolve o problema de caminhos de custo mínimo de única origem de forma mais geral, no qual os pesos podem ser negativos. Dado um grafo orientado ponderado $G=(V,E)$ com origem s e função peso $w = E \rightarrow R$, o algoritmo de Bellman-Ford retorna um valor booleano indicando se existe um ciclo de peso negativo acessível a partir da origem (CORMEN, 2002).

O algoritmo diminui progressivamente a estimativa $d[v]$ no peso de um caminho mais curto da origem s até cada vértice $v \in V$ até alcançar o peso real de caminho mais curto $\delta(s,v)$ (CORMEN, 2002).

Esse algoritmo, assim como o algoritmo de Dijkstra, inicializam todos os vértices com a estimativa de custo comum valor infinito e somente alteram este valor através do relaxamento. (CORMEN, 2002)

A diferença entre estes algoritmos está relacionada ao número de vezes em que o relaxamento é executado para cada aresta, no algoritmo de Dijkstra o relaxamento é executado uma vez para cada aresta e no algoritmo de Bellman-Ford cada aresta é relaxada muitas vezes. (CORMEN, 2002)

2.6.3 Algoritmo A Estrela (Algoritmo A*)

Alguns algoritmos de busca em grafos se guiam exclusivamente pela função heurística, a qual pode não indicar o caminho de custo mínimo, ou pelo custo real de se deslocar de um nó a outro, pode haver casos em que seja necessário realizar um movimento de custo maior para atingir a solução de custo mínimo (ROSSETTO, 2007).

Por isso um algoritmo de busca deveria levar em conta ambos os fatores, o valor heurístico dos nós e o custo real do percurso, quando aplicado a roteamento o custo real do percurso deve ser determinado traçando-se uma linha reta até o ponto de destino, pois este seria o menor caminho possível.

Assim, o algoritmo A* utiliza uma função de avaliação $f(n) = g(n) + h'(n)$, onde $h'(n)$ representa o valor heurístico do nó a avaliar desde o atual, n , até o final, e $g(n)$, o custo real do caminho percorrido para chegar ao dito nó n (ROSSETTO, 2007).

O algoritmo A* é uma combinação entre buscas do tipo primeiro em largura com primeiro em profundidade: enquanto $h'(n)$ tende a primeiro em profundidade, $g(n)$ tende a primeiro em largura. Deste modo, muda-se de caminho a cada vez que existe algum nó mais promissor.

3 ALGORITMO GENÉTICO (AG)

O algoritmo genético foi desenvolvido por Holland (1975) e Goldberg (1989) disseminou o uso de AG aplicando-o a uma série de problemas de otimização. Os AGs empregam um processo adaptativo de busca de soluções em problemas complexos, o que o torna uma técnica muito útil em problemas de otimização (RIBEIRO, 2005).

Os algoritmos genéticos são programas evolutivos baseados no princípio da seleção natural, onde os indivíduos mais aptos sobrevivem e os menos aptos tendem a serem descartados, a aptidão de um indivíduo é determinada através do cálculo da função objetivo, também conhecida por função *fitness* (DALBONI, 2003).

O primeiro passo de um AG é a geração da população inicial de cromossomos, esta população é formada por um conjunto aleatório de cromossomos que representam possíveis soluções do problema a ser resolvido (RIBEIRO, 2005).

Durante o processo evolutivo, a população é analisada e cada cromossomo recebe uma avaliação (aptidão) que reflete a qualidade da solução que ele representa (RIBEIRO, 2005).

Em geral, os cromossomos mais aptos são selecionados e os menos aptos são descartados. Os membros selecionados podem sofrer modificações em suas características fundamentais através dos operadores de cruzamento e mutação, gerando descendentes para a próxima geração. Este processo é repetido até que uma solução satisfatória seja encontrada (GRAÇA, 2009).

A figura 7 descreve de maneira sucinta os passos do algoritmo genético e as seguintes seções descrevem cada passo.

```

1:  $P \leftarrow$ População Inicial;
2: enquanto condição não satisfeita faça
3:    $P' \leftarrow$ Seleção( $P$ );
4:    $P \leftarrow$ Cruzamentos( $P'$ );
5:    $P \leftarrow$ Mutações( $P$ );
6: fim enquanto
7: Solução  $\leftarrow$ Melhor indivíduo( $P$ );

```

FIGURA 7- DESCRIÇÃO DOS PASSOS DO ALGORITMO GENÉTICO (OLIVEIRA, 2004)

3.1 Representação das soluções viáveis

Existem várias formas de representação possíveis para os cromossomos, como por exemplo: binária, inteira e real. A essa representação se dá o nome de alfabeto do algoritmo genético, a escolha da forma de representação depende do tipo de problema que se deseja resolver (MALAQUIAS, 2006).

Uma solução de um problema está associada a um cromossomo p representado na forma de um vetor com m posições: $p = (x_1, x_2, \dots, x_m)$ onde cada x_i representa um gene ou uma variável da solução (MALAQUIAS, 2006).

Dentre as representações conhecidas as mais utilizadas são a representação binária e por inteiros. A representação clássica de um AG é a representação binária, por ser mais facilmente interpretada e por se adaptar melhor aos mecanismos de renovação de uma população. Sendo assim os operadores genéticos utilizados devem ser definidos de acordo com a representação escolhida. (MALAQUIAS, 2006)

3.1.1 Representação Binária

Na representação binária os indivíduos são codificados por uma sequência de dígitos binários (0 e 1) e sua utilização está associada a algoritmos de codificação e decodificação, que permite converter a solução para a sequência binária e converter novamente para forma original (MALAQUIAS, 2006).

A representação binária pode ser aplicada em qualquer tipo de problema, mas nem sempre é a representação mais natural ou mais adequada para um problema. Pode ser o caso de problemas baseados em parâmetros numéricos, ou na ordem (MALAQUIAS, 2006).

3.1.2 Representação por inteiros

A representação por inteiros pode ser associada a solução de problemas de otimização combinatória, caracterizados pela busca de uma solução ótima para um conjunto finito de soluções. São exemplos o problema de menor caminho e o problema do caixeiro viajante (MALAQUIAS, 2006).

Para estes problemas a melhor representação de um indivíduo é um vetor de inteiros, na qual os inteiros representam uma ordem e cada inteiro representa um cliente a ser visitado. Considerando o cromossomo $p = (1, 3, 6, 5, 4, 2, 7, 8, 9)$ o vetor p representa uma solução de rota, representando a ordem na qual os clientes devem ser visitados (MALAQUIAS, 2006).

Este tipo de representação é interpretado de uma maneira peculiar e faz-se necessário a utilização de operadores genéricos específicos.

3.2 Função de custo

Cada possível solução é avaliada por uma função de custo que determina a qualidade desta solução, através do valor calculado pela função de custo é possível comparar as soluções disponíveis e escolher as melhores (MALAQUIAS, 2006).

Para o PRV a função de custo calcula a distância total de todos os roteiros incluídos no cromossomo e é também na função de custo que são avaliadas as restrições de capacidade de carga dos veículos.

A tabela 1 apresenta as distâncias entre os clientes determinada por uma linha reta entre as coordenadas de cada cliente. Neste exemplo o custo da solução (0 1 3 2 4 5 9 7 6 8) é dado por $d(0,1) + d(1,3) + d(3,2) + d(2,4) + d(4,5) + d(5,9) + d(7,6) + d(6,8) + d(8,0)$

TABELA 1 - DISTÂNCIA ENTRE OS CLIENTES

	1	2	3	4	5	6	7	8	9	10
1	0,00	4,76	4,12	2,00	5,66	5,20	2,24	4,61	3,00	4,12
2	4,76	0,00	4,86	0,00	8,50	9,39	6,96	65,40	7,05	8,43
3	4,12	4,86	0,00	2,24	3,93	5,80	5,10	8,50	7,07	8,00
4	2,00	3,83	2,24	0,00	4,90	55,70	3,61	6,26	5,00	6,08
5	5,66	8,50	3,93	4,90	0,00	3,03	4,88	10,15	7,68	7,96
6	5,20	9,39	5,80	55,70	3,03	0,00	3,35	8,92	6,00	5,80
7	2,24	6,96	5,10	3,61	4,88	3,35	0,00	5,59	2,83	3,16
8	4,61	65,40	8,50	6,26	10,15	8,92	5,59	0,00	3,04	4,03
9	3,00	7,05	7,07	5,00	7,68	6,00	2,83	3,04	0,00	1,41
10	4,12	8,43	8,00	6,08	7,96	5,80	3,16	4,03	1,41	0,00

3.3 Função de aptidão

A função de aptidão de um indivíduo é uma maneira de avaliar a qualidade de solução que este representa. Ao longo do processo evolutivo, cada membro da população é avaliado e recebe uma nota ou índice, que determina a aptidão deste perante o problema em que está sendo avaliado. Esta nota é conhecida por aptidão ou *fitness* (DALBONI, 2003).

A diferença entre as funções é que a função de aptidão tende a ser maximizada e a função de custo minimizada, sendo assim seja $f(x)$ a função de custo definida na seção 3.2, podemos definir a função de aptidão como sendo: $g(x) = -f(x) + f_{max}$ onde f_{max} é o maior valor da função de custo para a população sendo avaliada (MALAQUIAS, 2006).

Os AGs procuram preservar os indivíduos de mais alta aptidão e eliminar os com menores aptidão (DALBONI, 2003).

3.4 População inicial

A população inicial é formada por um conjunto de cromossomos que representam possíveis soluções do problema a ser resolvido, esta população pode ser gerada aleatoriamente ou utilizando algoritmos heurísticos (GRAÇA, 2009)

Assim que gerada a população inicial é avaliada pela função de aptidão a fim de medir sua qualidade. (MALAQUIAS, 2006)

3.5 Seleção

O processo de seleção (ou reprodução) é responsável por escolher quais indivíduos serão submetidos às operações genéticas como cruzamento e mutação e os indivíduos resultantes destas operações irão compor a nova população. (SOARES, 2007)

A seleção não se deve basear unicamente na escolha do melhor indivíduo, pois há a possibilidade deste não estar próxima da solução ótima global. Por isso deve-se manter alguma chance de que indivíduos com aptidão relativamente baixa participem do processo de reprodução (MOGNON, 2004)

As estratégias de seleção podem ser classificadas como estocásticas ou determinísticas (MOGNON, 2004) e as seções seguintes apresentam algumas destas estratégias.

3.5.1 Dizimação

Uma estratégia determinística consiste em ordenar os indivíduos de acordo com o resultado da função de aptidão e simplesmente remover um número fixo de indivíduos que apresentarem baixa aptidão, ou seja, criar um patamar e eliminar aqueles que estiverem abaixo deste patamar. Os pais então serão escolhidos de forma aleatória entre os indivíduos que sobreviveram ao processo de dizimação (MOGNON, 2004).

A vantagem desta estratégia de seleção é a simplicidade de implementação, que consiste em determinar quais indivíduos possuem aptidão suficiente para permanecer na população. A desvantagem é que características genéticas únicas podem ser perdidas uma vez que o indivíduo é removido da população (MOGNON, 2004).

3.5.2 Seleção proporcional

Um dos métodos mais populares métodos estocásticos de seleção é a seleção proporcional, também conhecida como roda da roleta. Neste método, os indivíduos são selecionados com base na probabilidade de seleção que é proporcional à função objetivo (MOGNON, 2004).

A estratégia de seleção proporcional pode ser interpretada como uma roleta onde cada indivíduo da população é representado em uma porção proporcional ao seu índice de aptidão (MOGNON, 2004).

A figura 8 ilustra as probabilidades de seleção para uma população que utiliza o vetor de aptidão (0.03, 0.05, 0.12, 0.15, 1.00)

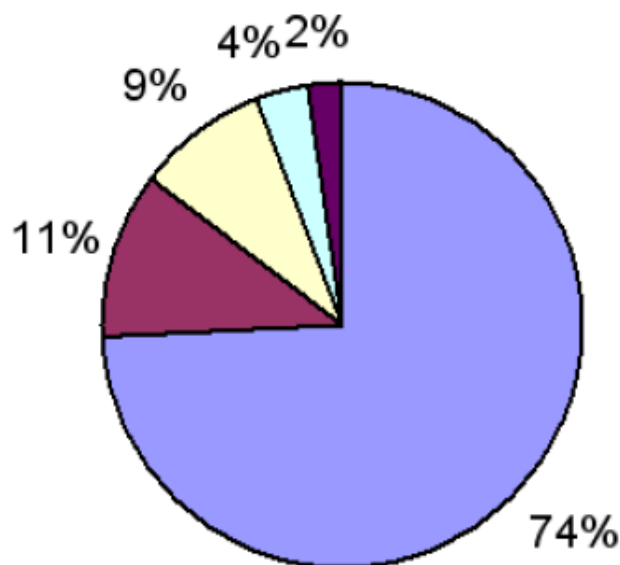


FIGURA 8 - GRÁFICO DE PROBABILIDADE DE SELEÇÃO COM O MÉTODO DE SELEÇÃO PROPORCIONAL (MOGNON, 2004)

Desta forma uma porção maior é atribuída ao indivíduo com alta aptidão, a roleta é girada o número de vezes necessário para obter o número de pares de indivíduos requeridos para a execução dos operadores genéticos (MOGNON, 2004).

A vantagem desta estratégia é que todos os indivíduos têm a oportunidade de serem selecionados e esta é proporcional a sua aptidão (MOGNON, 2004).

Esta estratégia tende a sofrer o efeito de dominância se houver algum indivíduo com alta aptidão em relação à média, podendo levar o algoritmo genético a estagnação, pois a seleção está implicitamente relacionada com a diversidade da população (MOGNON, 2004).

3.5.3 Torneio

Nesta estratégia de seleção é criado aleatoriamente um subconjunto de N indivíduos, os indivíduos deste subconjunto competem entre si com base na sua aptidão e somente o vencedor será utilizado, após a escolha todos os indivíduos são devolvidos a população e um novo subconjunto é selecionado, este processo é repetido até ser selecionado o número requerido de indivíduos para a aplicação dos operadores genéticos (MOGNON, 2004).

A figura 9 apresenta o funcionamento desta estratégia, a implementação desta estratégia é relativamente simples e requer baixo custo computacional, sendo um dos motivos de sua popularidade (MOGNON, 2004).

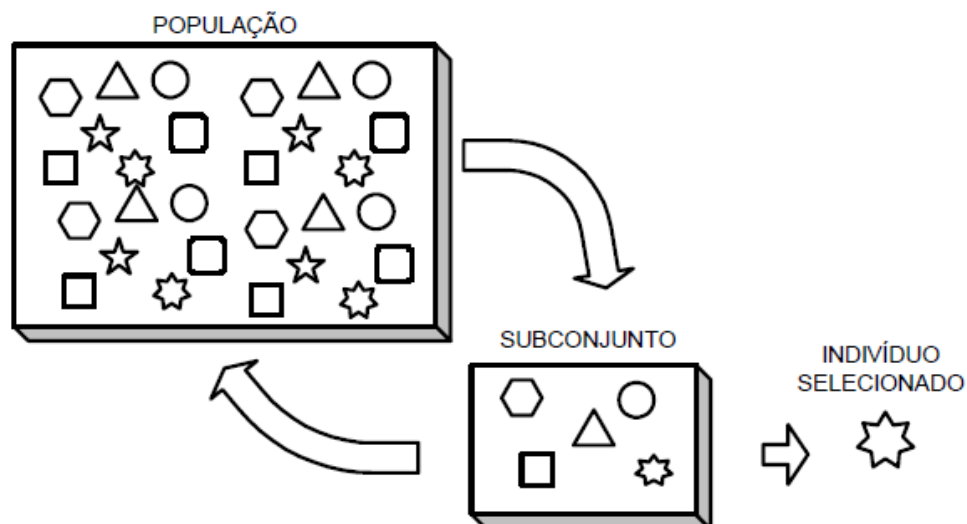


FIGURA 9 - SELEÇÃO PELO MÉTODO DE TORNEIO (MOGNON, 2004)

3.6 Operadores genéticos

A função dos operadores genéticos é definir regras para uma renovação eficaz de uma população, são responsáveis por criar novos pontos de busca no espaço de solução, com base nos elementos da população atual. Esses procedimentos ocorrem de forma iterativa tentando melhorar a qualidade de cada indivíduo (MALAQUIAS, 2006).

Os operadores genéticos são necessários para que a população se diversifique, pois após um determinado número de gerações, as soluções encontradas tendem a ficar similares.

O desenvolvimento de operadores genéticos está relacionado ao tipo de representação da solução do problema na forma de um cromossomo conforme explicado na seção 2.6.1.

Os operadores mais utilizados são: (MALAQUIAS, 2006)

1. Cruzamento;

2. Mutação;

3.6.1 Cruzamento (*Crossover*)

A primeira etapa na formação de uma nova população é chamada de cruzamento, na qual é gerada uma nova população a partir da população atual, esta operação é realizada através da escolha de pares de indivíduos (os pais) que serão usados para gerar um par de filhos, estes filhos apresentarão características genéticas herdadas dos pais. (MALAQUIAS, 2006)

Dentre os operadores de cruzamento já propostos, alguns serão apresentados nas subseções seguintes e maiores detalhes pode ser encontrados em (LARRAÑAGA, 1999).

3.6.2 Cruzamento de mapeamento parcial (PMX)

O operador de cruzamento de mapeamento parcial (*partially-mapped crossover* – PMX) transfere informações de ordem e de posição das rotas dos pais para as rotas dos filhos. Uma parte da sequência de um pai é mapeada a uma parte da sequência do outro pai e é preservada no filho, o restante das informações é trocada entre os pais (MALAQUIAS, 2006).

Utilizando-se como exemplo as sequências (1 2 3 4 5 6 7 8 9) como a rota do *pai1* e (3 7 5 1 6 8 2 4) como a rota do *pai2* (MALAQUIAS, 2006).

O operador PMX primeiro seleciona aleatoriamente dois pontos de corte. Suponha que o primeiro ponto de corte seja selecionado entre o terceiro e o quarto elementos, e o segundo entre o sexto e o sétimo elementos de cada sequência pai. Estas subsequências são chamadas de seções de mapeamento, a figura 10 exemplifica a seção de mapeamento do exemplo utilizado.

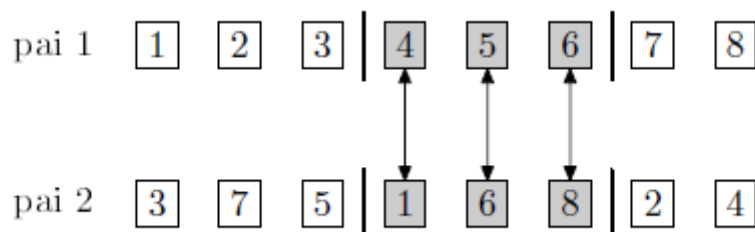


FIGURA 10 - EXEMPLO DA SEÇÃO DE MAPAAMENTO DO PMX (MALAQUIAS, 2006)

Em seguida, os elementos da seção de mapeamento do *pai1* são copiados para o *filho2* e os elementos da seção de mapeamento do *pai2* são copiados para o *filho1* conforme exemplificado na figura 11.

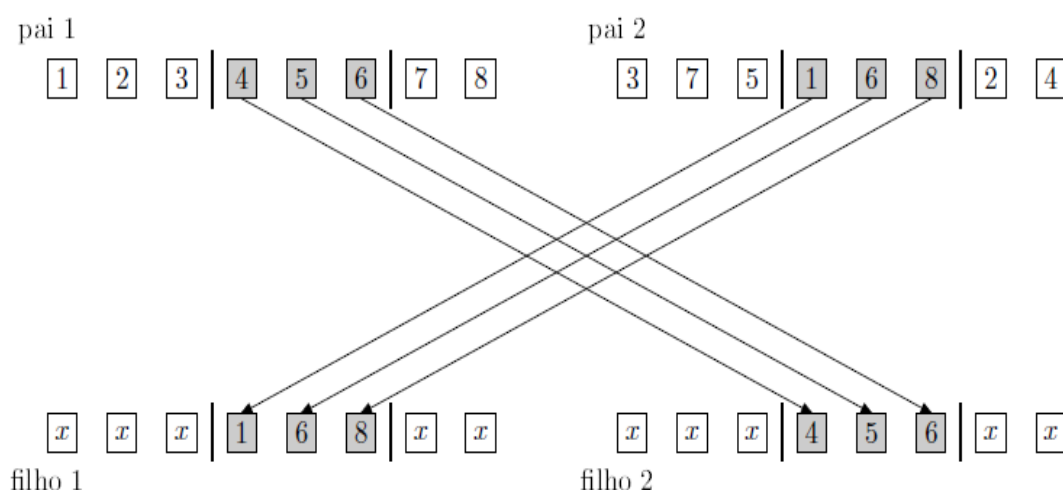


FIGURA 11 - CÓPIA DOS ELEMENTOS DA SEÇÃO DE MAPAAMENTO DOS PAIS PARA OS FILHOS (MALAQUIAS, 2006)

O próximo passo é copiar o restante dos elementos do *pai1* para o *filho1*, em caso do elemento já existir no *filho1* é necessário localizar a posição em que este está inserido e localizar o elemento correspondente a esta posição no *pai1*, este procedimento de localizar o elemento na posição correspondente no pai deve ser feito até que seja encontrado um elemento que não exista no filho para ser inserido. (MALAQUIAS, 2006).

Esta operação é realizada para todos os elementos restantes no *filho1* e repetida para os elementos do *filho2* copiando os elementos do *pai2*. A figura 12 explica como são realizadas as operações de mapeamento e cópia dos elementos. (MALAQUIAS, 2006)

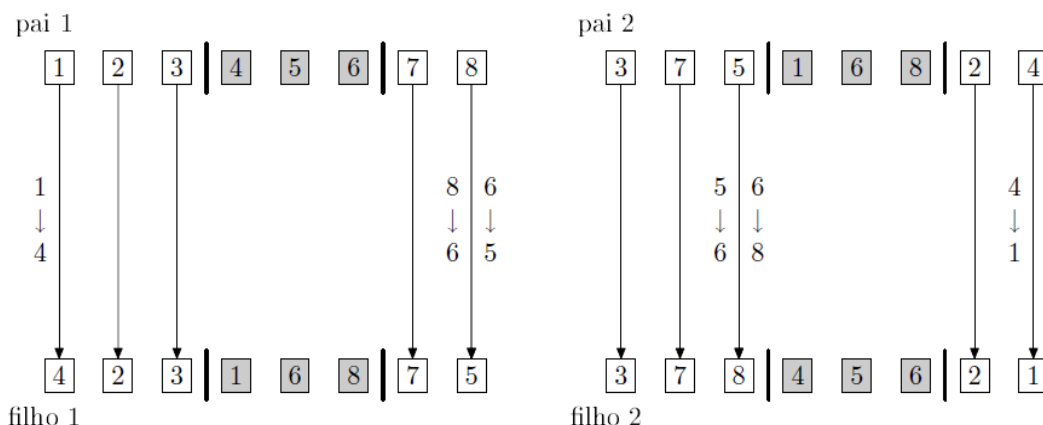


FIGURA 12 - CÓPIA DOS ELEMENTOS RESTANTES NO PAI E A TROCA PELOS ELEMENTOS DA SEÇÃO DE MAPEAMENTO CORRESPONDENTE (MALAQUIAS, 2006)

3.6.3 Cruzamento de ponto único

Neste operador de cruzamento é necessário gerar um ponto de corte e a partir deste ponto copiar o que está à esquerda do primeiro pai para o primeiro filho e completar o primeiro filho com o que está à direita do ponto de corte do segundo pai e

para o segundo inverter o processo, copiar a primeira parte do segundo pai e a segunda do primeiro pai (DALBONI, 2003)

Utilizando como exemplo a figura 13, as sequências dos pais como $p1$ e $p2$ e as sequências dos filhos $o1$ e $o2$ e o ponto de corte sendo entre os elementos 3 e 4 (DALBONI, 2003).

$$\begin{aligned} p_1 &= (0\ 1\ 2\ |3\ 4\ 5\ 6) = (p_{11}|p_{12}) & \text{e} & \quad o_1 = (0\ 1\ 2\ |6\ 5\ 4\ 3) = (p_{11}|p_{22}) \\ p_2 &= (0\ 2\ 1\ |6\ 5\ 4\ 3) = (p_{21}|p_{22}) & \text{e} & \quad o_2 = (0\ 2\ 1\ |3\ 4\ 5\ 6) = (p_{21}|p_{12}) \end{aligned}$$

FIGURA 13 - OPERADOR CRUZAMENTO DE UM ÚNICO PONTO (DALBONI, 2003)

Notamos que a parte esquerda do indivíduo $p1$ se uniu com a parte direita do indivíduo $p2$ para formar o indivíduo filho $o1$, que possui as características combinadas de ambos os pais (DALBONI, 2003).

3.6.4 Cruzamento de ordem (OX)

O operador de cruzamento de ordem explora a propriedade de representação do caminho, em que ordem é importante e não a posição. Ele escolhe um subroteiro de um dos pais preservando a ordem dos elementos do outro pai (MALAQUIAS, 2006).

Considere os indivíduos da figura 14 e suponha que o primeiro ponto de corte seja entre o segundo e terceiro elemento e o segundo ponto de corte entre o quinto e o sexto elemento.

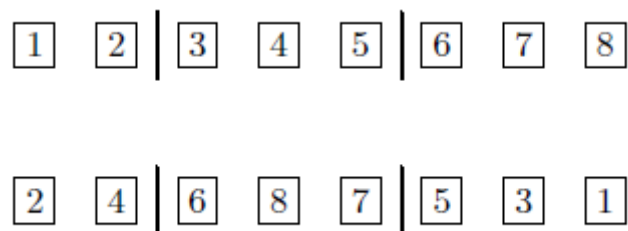


FIGURA 14 - EXEMPLO DE INDIVÍDUOS PARA O OPERADOR DE CRUZAMENTO DE ORDEM

Os filhos são criados seguindo os seguintes passos: primeiro, são copiados os elementos dentro dos pontos de corte para os filhos, o resultado é apresentado na figura 15.

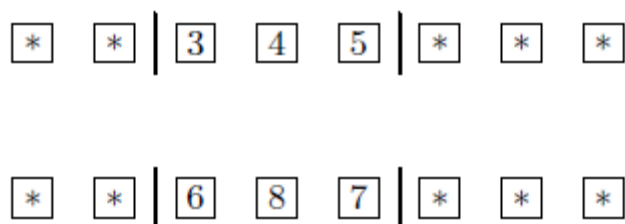


FIGURA 15- CÓPIA DOS ELEMENTOS DOS PAIS PARA OS FILHOS NO OPERADOR DE CRUZAMENTO DE ORDEM (MALAQUIAS, 2006)

O segundo passo é copiar a partir do segundo ponto de corte os elementos do outro pai à medida que eles aparecem omitindo os que já existem no filho. A figura 16 apresenta o resultado da execução deste operador.

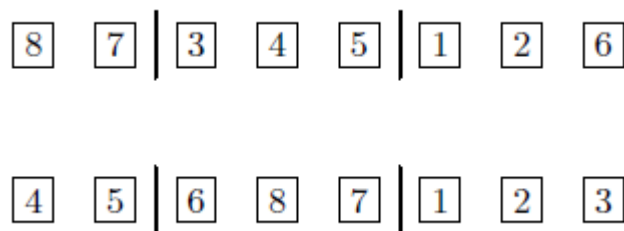


FIGURA 16- RESULTADO DA EXECUÇÃO DO OPERADOR DE CRUZAMENTO DE ORDEM
(MALAQUIAS, 2006)

3.7 Mutação

A operação de mutação é utilizada para garantir uma maior varredura do espaço e evitar que o algoritmo genético se aproxime muito rapidamente para mínimos locais.

A mutação é efetuada alterando-se o valor de um gene de um indivíduo selecionado aleatoriamente com uma determinada probabilidade, denominada probabilidade de mutação, ou seja, vários indivíduos da nova população podem ter um de seus genes alterado aleatoriamente.

3.7.1 Mutação por troca (EM)

O operador de mutação por troca (*Exchange mutation* - EM) aleatoriamente seleciona duas posições no indivíduo e troca suas posições. Considere o indivíduo (1 2 3 4 5 6 7 8) e suponha que as posições selecionadas sejam a terceira e quinta. A mutação resulta no indivíduo (1 2 5 4 3 6 7 8), a figura 17 explica o funcionamento deste processo (MALAQUIAS, 2006).

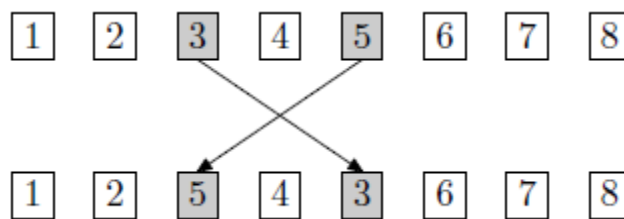


FIGURA 17- MUTAÇ O POR TROCA (MALAQUIAS, 2006)

3.7.2 Muta o por invers o simples (SIM)

O operador de muta o por invers o simples (*simple inversion operator*) seleciona aleatoriamente dois pontos de corte e inverte os elementos do subconjunto formado a partir dos pontos de corte (MALAQUIAS, 2006).

Considerando a sequ ncia (1 2 3 4 5 6 7 8), suponha que o primeiro ponto de corte selecionado seja entre o primeiro e segundo elemento e o segundo entre o quinto e o sexto elemento, a muta o resulta na sequ ncia (1 5 4 3 2 6 7 8), a figura 18 apresenta o funcionamento deste operador (MALAQUIAS, 2006).

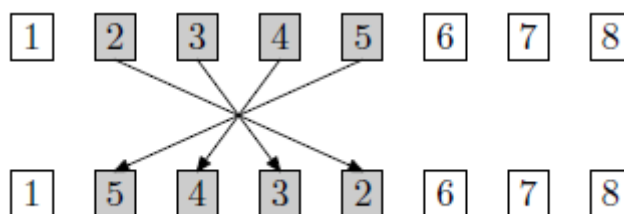


FIGURA 18- MUTA O POR INVERS O SIMPLES (MALAQUIAS, 2006)

3.7.3 Mutação por deslocamento (DM)

O operador de mutação por deslocamento (*DisplacementMutation*), também conhecido por mutação por corte, primeiro seleciona aleatoriamente uma subsequência e esta é removida de sua sequência e inserida em um ponto aleatoriamente selecionado (MALAQUIAS, 2006).

Considere como exemplo a sequência (1 2 3 4 5 6 7 8) e suponha que a subsequência (3 4 5) seja selecionada. Após a remoção da subsequência tem-se (1 2 6 7 8) (MALAQUIAS, 2006).

Supondo que o ponto de corte selecionado seja a posição 7 temos a seguinte sequência após a inserção (1 2 6 7 3 4 5 8). A figura 19 apresenta o funcionamento deste operador.

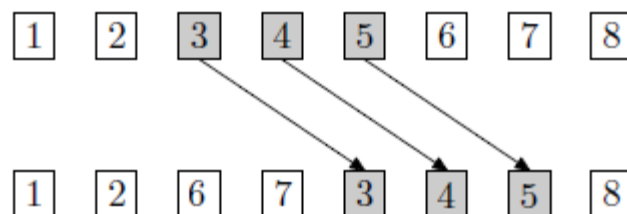


FIGURA 19 - MUTAÇÃO POR DESLOCAMENTO (DM)

3.7.4 Mutação por inserção (ISM)

O operador de mutação por inserção escolhe aleatoriamente um elemento na sequência, remove este elemento e o insere em uma posição escolhida aleatoriamente (MALAQUIAS, 2006).

Considere como exemplo a sequência (1 2 3 4 5 6 7 8) e suponha que o elemento na posição quatro seja selecionado e seja escolhida a posição sete para este ser inserido, a figura 20 apresenta o funcionamento deste operador (MALAQUIAS, 2006).

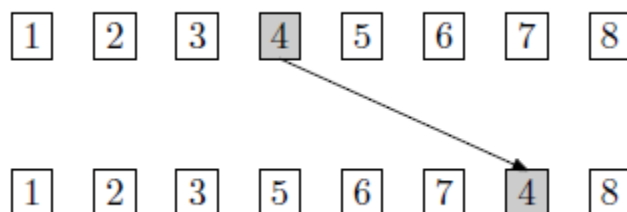


FIGURA 20 - MUTAÇÃO POR INSERÇÃO (ISM) (MALAQUIAS, 2006)

3.7.5 Mutação por inversão (IVM)

O operador de mutação por inversão é similar ao operador por deslocamento, ele também seleciona aleatoriamente uma subsequência, remove ela da sequência e a insere em uma posição selecionada aleatoriamente. Sua diferença em relação ao operador por deslocamento é que a subsequência é inserida em ordem reversa (MALAQUIAS, 2006).

Considere como exemplo a sequência (1 2 3 4 5 6 7 8) e suponha que a subsequência (3 4 5) seja selecionada. Após a remoção da subsequência tem-se (1 2 6 7 8), suponha que a posição selecionada seja a sétima posição, a subsequência é inserida resultando na sequência (1 2 5 4 3 6 7 8), a figura 21 apresenta o funcionamento deste operador (MALAQUIAS, 2006).

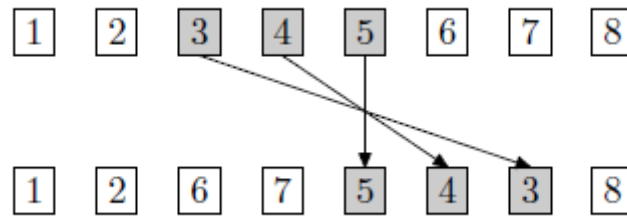


FIGURA 21- MUTAÇÃO POR INVERSÃO (IVM) (MALAQUIAS, 2006)

3.7.6 Scramble Mutation (SM)

O operador de mutação *scramble* seleciona aleatoriamente uma subsequência e mistura os elementos dentro dela. Considere como exemplo a sequência (1 2 3 4 5 6 7 8) e supondo que seja escolhida a subsequência (4 5 6 7) o resultado é mostrado na figura 22 (MALAQUIAS, 2006).

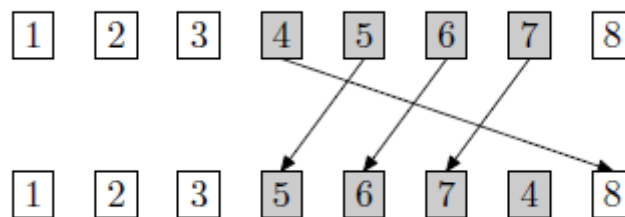


FIGURA 22 - OPERADOR DE MUTAÇÃO SCRAMBLE (SM) (MALAQUIAS, 2006)

4 PROPOSTA DE SOLUÇÃO

Este capítulo propõe um protótipo para a solução do problema de roteamento de veículos clássico utilizando um algoritmo genético respeitando as restrições de capacidade de cada veículo.

No problema a ser resolvido existe um centro de distribuição, vários pontos de entrega (clientes) e a frota de veículos, no qual cada cliente possui a quantidade de produtos que devem ser entregues e para a frota de veículos todos possuem a mesma capacidade.

Na prática, as distâncias entre os pontos de entrega devem ser calculadas considerando o trajeto entre eles, para simplificar este protótipo serão consideradas as distâncias lineares calculadas em um sistema de coordenadas cartesianas, onde cada cliente i é representado por suas coordenadas $(x_i \text{ e } y_i)$. A distância entre dois pontos de entrega i e j é dada por $\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$

Para resolução deste problema usando algoritmos genéticos é necessário definir uma representação para as soluções (cromossomos), a função de custo a ser minimizada, o método de seleção e os operadores genéticos (cruzamento e mutação) a serem utilizados na geração de novas populações.

4.1 Representação de uma solução

Alguns problemas de otimização envolvem a classificação de uma lista para a obtenção da solução ótima. Para um grande número de problemas tratados até mesmo com técnicas heurísticas de Inteligência Artificial esta lista é uma permutação e para

obter a solução ótima é necessário buscar a melhor permutação de seus elementos (MALAQUIAS, 2006).

Conforme apresentado na seção 2.7 a representação de uma solução está devidamente relacionada ao tipo de problema a ser resolvido, para a resolução do PRV clássico a representação que mais se adapta a este problema é a representação por inteiros (seção 2.7.2).

Neste caso a lista é a sequência de clientes a serem visitados partindo do depósito, visitando os clientes e retornando ao depósito. A figura 23 apresenta um exemplo de solução para um problema com nove pontos de entrega.

veículo	roteiro
v_1	6 – 5 – 2
v_2	4 – 1 – 9 – 3
v_3	7 – 8

FIGURA 23 - EXEMPLO DE SOLUÇÃO PARA UM PROBLEMA COM NOVE PONTOS DE ENTREGA (MALAQUIAS, 2006)

Para a utilização de um algoritmo genético o primeiro passo é representar uma solução como um cromossomo. Para este problema o cromossomo é representado por uma lista circular de números inteiros, onde cada número inteiro identifica um ponto no roteiro (o centro de distribuição ou um ponto de entrega). Para o exemplo da figura 20 tem-se o cromossomo apresentado na figura 24.

0 – 6 – 5 – 2 – 0 – 3 – 4 – 1 – 9 – 0 – 7 – 8 – 0

FIGURA 24- EXEMPLO DE CROMOSSOMO

O roteiro de um veículo começa no 0 e termina no 0 seguinte, uma vez que o 0 representa o centro de distribuição, e a partir deste cromossomo pode-se extrair o roteiro apresentado na figura 25, este roteiro pode ser atendido por três veículos ou um veículo com capacidade limitada retornando ao centro de distribuição.

$$0 - 6 - 5 - 2 - 0$$

$$0 - 3 - 4 - 1 - 9 - 0$$

$$0 - 7 - 8 - 0$$

FIGURA 25 - ROTA EXTRAÍDA DE UM CROMOSSOMO

Como a representação é realizada por uma permutação de n números naturais, uma restrição na representação adotada é a não repetição de um número na lista circular, porém o número que representa o depósito pode ocorrer diversas vezes, para adaptar a representação a esta restrição é necessário introduzir novo pontos representando o centro de distribuição, um para cada veículo adicional. As coordenadas destes pontos devem ser iguais as coordenadas do centro de distribuição, sendo assim a solução apresentada na figura 24 passa a ser representada pelo cromossomo da figura 26. Na qual os números 0, 10 e 11 representam o mesmo centro de distribuição.

$$0 - 6 - 5 - 2 - 10 - 3 - 4 - 1 - 9 - 11 - 7 - 8$$

FIGURA 26- CROMOSSOMO ADAPTADO PARA NÃO REPETIR O CENTRO DE DISTRIBUIÇÃO
(MALAQUIAS, 2006)

4.2 Modelagem da solução proposta para a resolução do PRV clássico

A figura 27 apresenta um diagrama de classes da solução proposta para resolução do problema do roteamento de veículos clássico através da utilização de um algoritmo genético.

Foram implementados os métodos de seleção apresentados na seção 3.5, os operadores de cruzamento apresentados na seção 3.6 e os operadores de mutação apresentados na seção 3.7.

O *enum* *Selecao* é usado para informar ao algoritmo qual método de seleção foi selecionado pelo usuário e o mesmo acontece com os *enums* *Cruzamento* e *Mutacao* para os operadores de mutação e cruzamento respectivamente.

O objeto *Cliente* representa o ponto de entrega, os clientes são lidos a partir de um arquivo de texto que possui sua identificação, coordenada e quantidade de produtos necessária.

Para a execução do algoritmo é necessário determinar alguns parâmetros como taxa de mutação, taxa de cruzamento, tamanho da população, número de gerações e capacidade dos veículos.

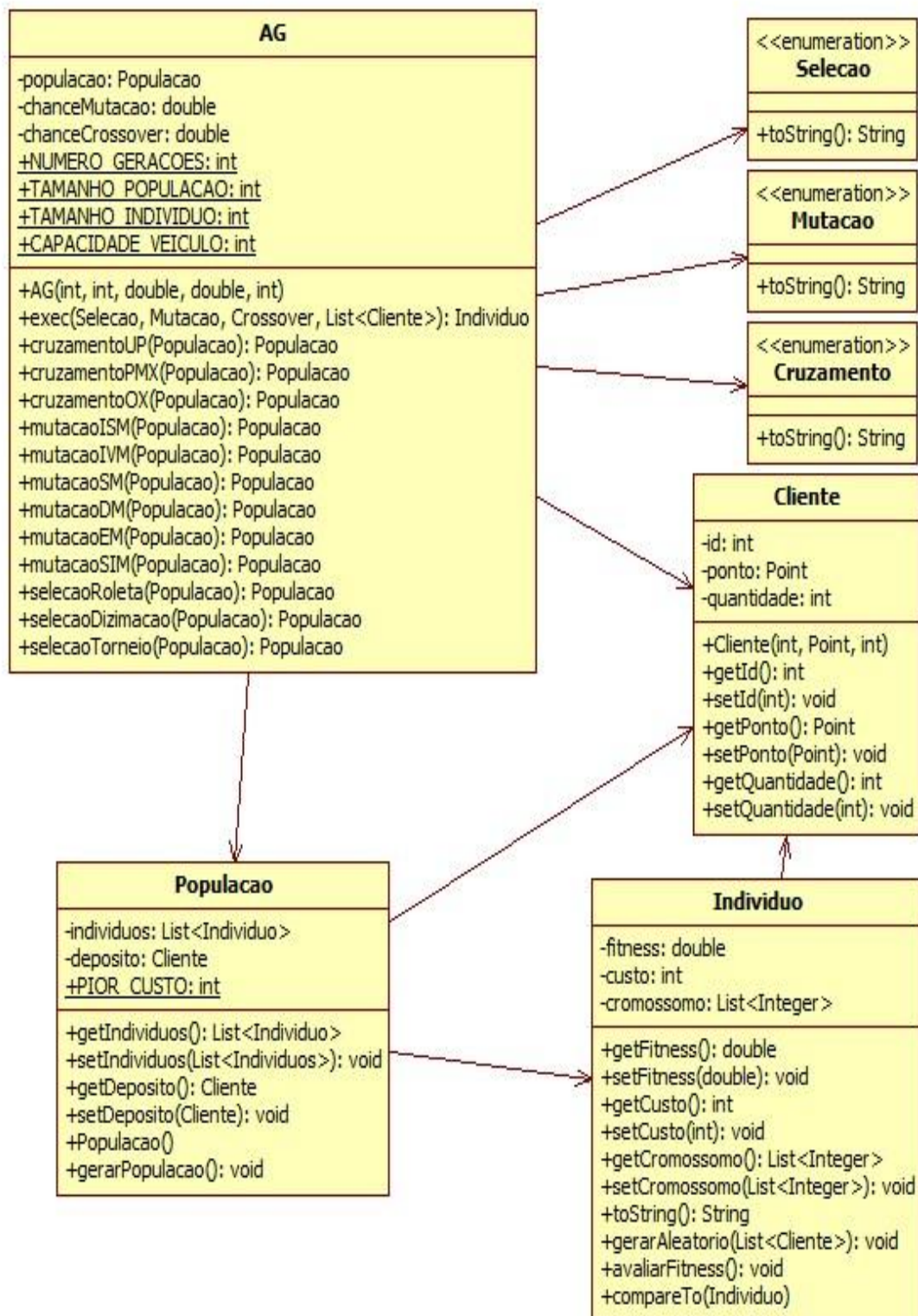


FIGURA 27- DIAGRAMA DE CLASSES DA SOLUÇÃO PROPOSTA PARA O PRV CLÁSSICO

A figura 28 apresenta o sistema em execução para a instância E-n51-k5 (CHRISTOFIDES, 1969) com 50 clientes e um depósito, foram geradas 5 rotas e cada uma é representada por uma cor diferente.

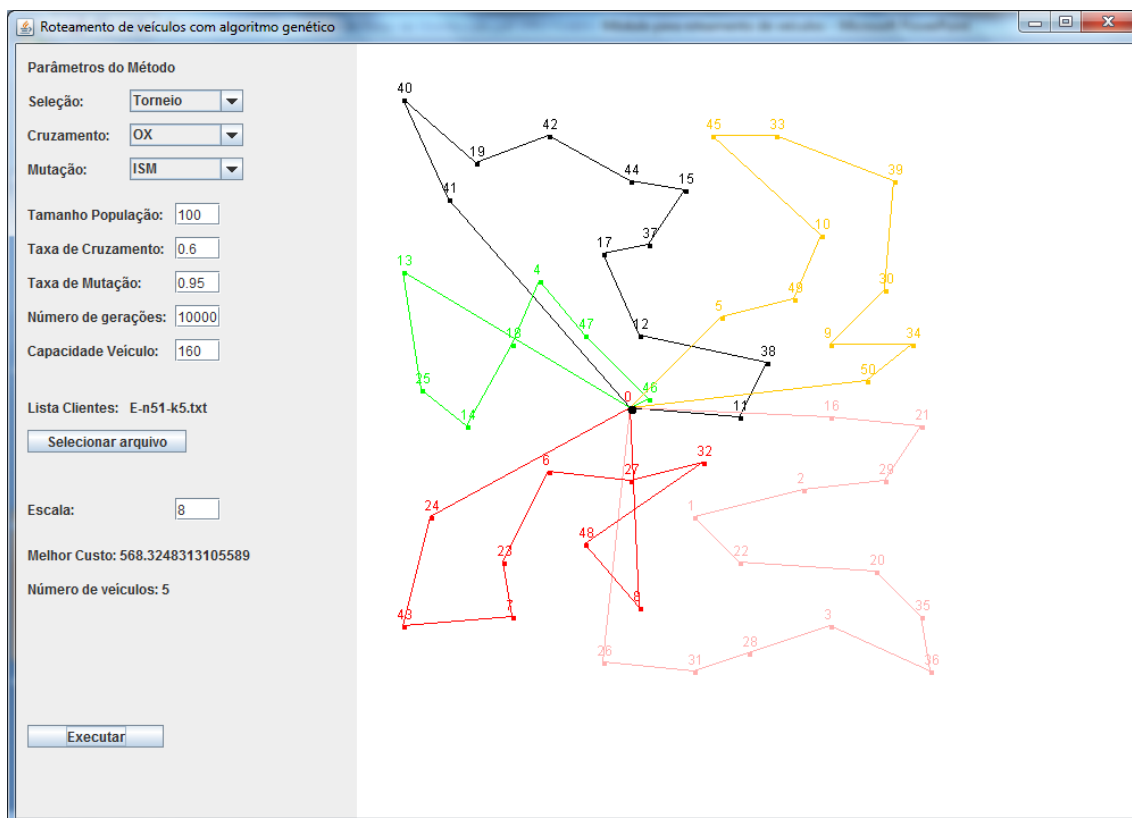


FIGURA 28 – TELA DE EXECUÇÃO DO PROTÓTIPO

5 RESULTADOS COMPUTACIONAIS

Neste capítulo são apresentados os problemas usados para teste e validação do algoritmo proposto e os resultados obtidos a partir da solução proposta.

5.1 Problema teste

Para teste do sistema foi utilizada a instância E-n51-k5 (CHRISTOFIDES, 1969), os dados referentes ao depósito são apresentados na tabela 2 e os dados referentes aos clientes são apresentados na tabela 3, neste caso são 50 clientes com demandas diferentes, com relação à frota de veículos foi considerado que a frota é homogênea, ou seja, todos os veículos possuem a mesma capacidade e esta é determinada pelo usuário no sistema.

Nos testes realizados o programa foi executado 10 vezes para cada situação e foram anotados o custo mínimo e o custo médio obtidos nestas execuções. A tabela 4 apresenta os valores padrões utilizados na execução quando o parâmetro não era o objeto de estudo.

Por exemplo, o tamanho da população utilizado em todos os testes foi de 100 indivíduos exceto no caso em que o tamanho da população era o objeto testado como apresenta o gráfico comparativo apresentado na figura 32.

TABELA 2- COORDENADAS DO DEPÓSITO

Depósito	Coordenadas	
	X	Y
0	30	40

TABELA 3 - COORDENADAS DOS CLIENTES

Cliente	Coordenadas		Demanda
	X	Y	
1	37	52	7
2	49	49	30
3	42	64	16
4	20	26	9
5	40	30	21
6	21	47	15
7	17	63	19
8	31	62	23
9	52	33	11
10	51	21	5
11	42	41	19
12	31	32	29
13	5	25	23
14	12	42	21
15	36	16	10
16	52	41	15
17	27	23	3
18	17	33	41
19	13	13	9
20	57	58	28
21	62	42	8
22	42	57	8
23	16	57	16
24	8	52	10

25	7	38	28
26	27	61	7
27	30	48	15
28	43	67	14
29	58	48	6
30	58	27	19
31	37	60	11
32	38	46	12
33	46	10	23
34	61	33	26
35	62	53	17
36	63	59	6
37	32	22	9
38	45	35	15
39	59	15	14
40	5	6	7
41	10	17	27
42	21	10	13
43	5	64	11
44	30	15	16
45	39	10	10
46	32	39	5
47	25	32	25
48	25	55	17
49	48	28	18
50	56	37	10

TABELA 4 - PARÂMETROS UTILIZADOS COMO PADRÃO

Parâmetro	Padrão
Operador de cruzamento	Cruzamento de ordem (OX)
Operador de mutação	Deslocamento (DM)
Taxa de cruzamento	60%
Taxa de mutação	95%
Tamanho da população	100
Número de gerações	10000
Capacidade dos veículos	160

5.2 Parâmetros do algoritmo genético

Todos os testes foram executados no seguinte ambiente de teste

- 4Gb RAM
- Intel Core i5 2.53 GHz
- Windows 7

Os gráficos a seguir apresentam os resultados obtidos nos testes realizados para analisar a influência dos parâmetros os resultados obtidos com o algoritmo genético

A figura 29 apresenta os métodos de seleção, observa-se que o método da roleta apresentou melhores resultados em relação aos demais.

A figura 30 apresenta a comparação entre os operadores de mutação, observa-se que os operadores de mutação SIM e ISM apresentaram melhores resultados.

A figura 31 apresenta a comparação entre os operadores de cruzamento, observa-se que o operador OX apresentou os melhores resultados.

A figura 32 apresenta uma comparação entre os resultados obtidos com populações de tamanhos variados, observa-se que os melhores resultados foram obtidos com as populações maiores.

A figura 33 apresenta uma comparação entre diferentes taxas de mutação, observa-se que os melhores resultados foram obtidos com as taxas entre 60% e 90%. É possível observar que os operadores de mutação exercem grande influência no resultado.

A figura 34 apresenta uma comparação entre diferentes taxas de cruzamento, observa-se que os melhores resultados foram obtidos com as maiores taxas mas que o cruzamento não tem grande influência no resultado final.

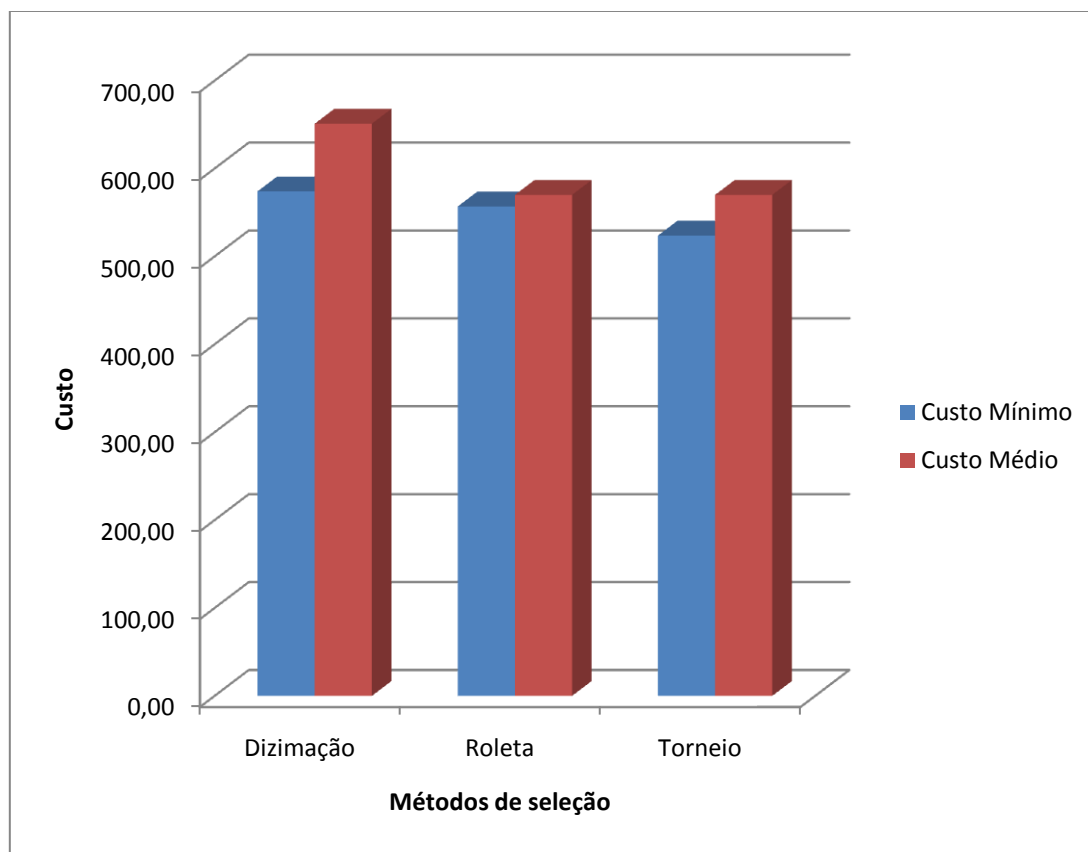


FIGURA 29 - GRÁFICO COMPARATIVO ENTRE OS MÉTODOS DE SELEÇÃO

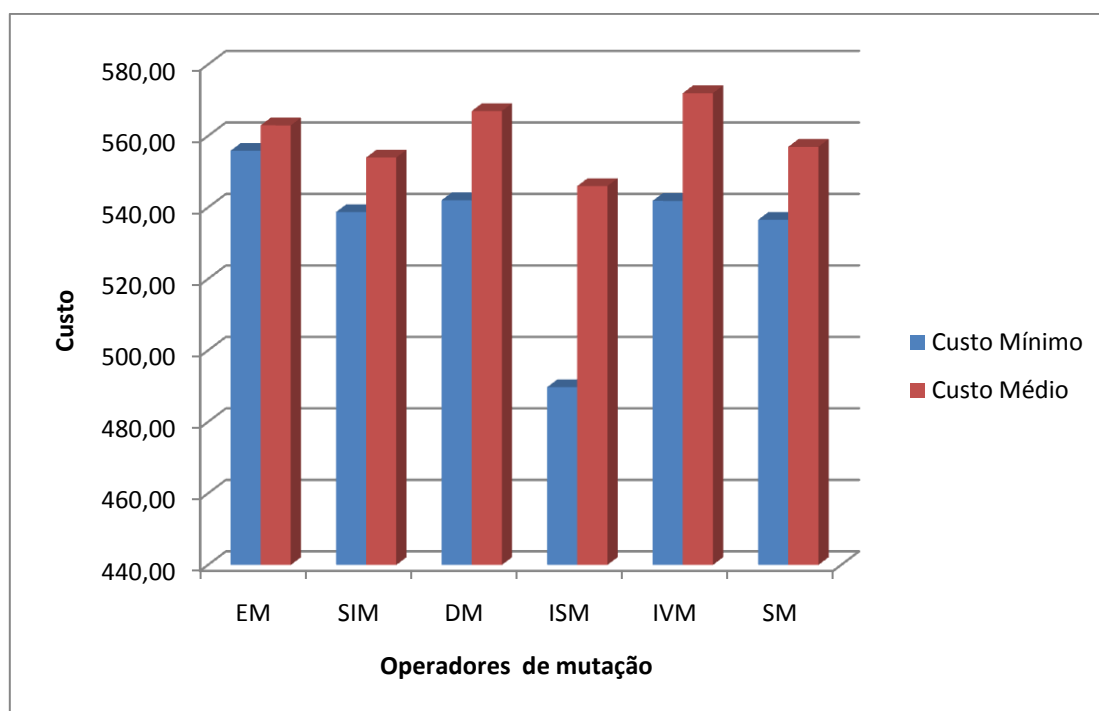


FIGURA 30- GRÁFICO COMPARATIVO ENTRE OS OPERADORES DE MUTAÇÃO

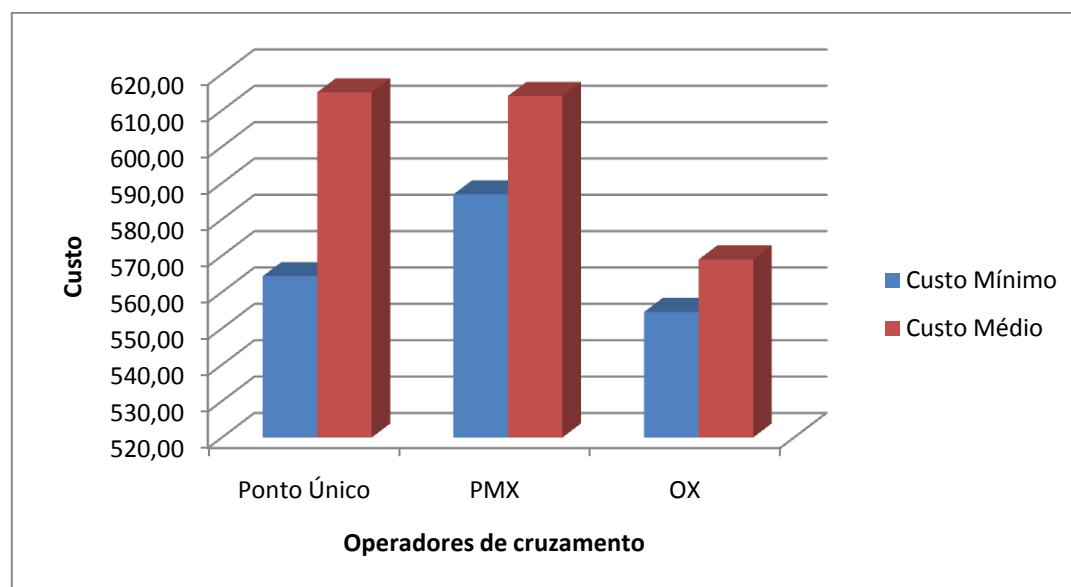


FIGURA 31 - GRÁFICO COMPARATIVO ENTRE OS OPERADORES DE CRUZAMENTO

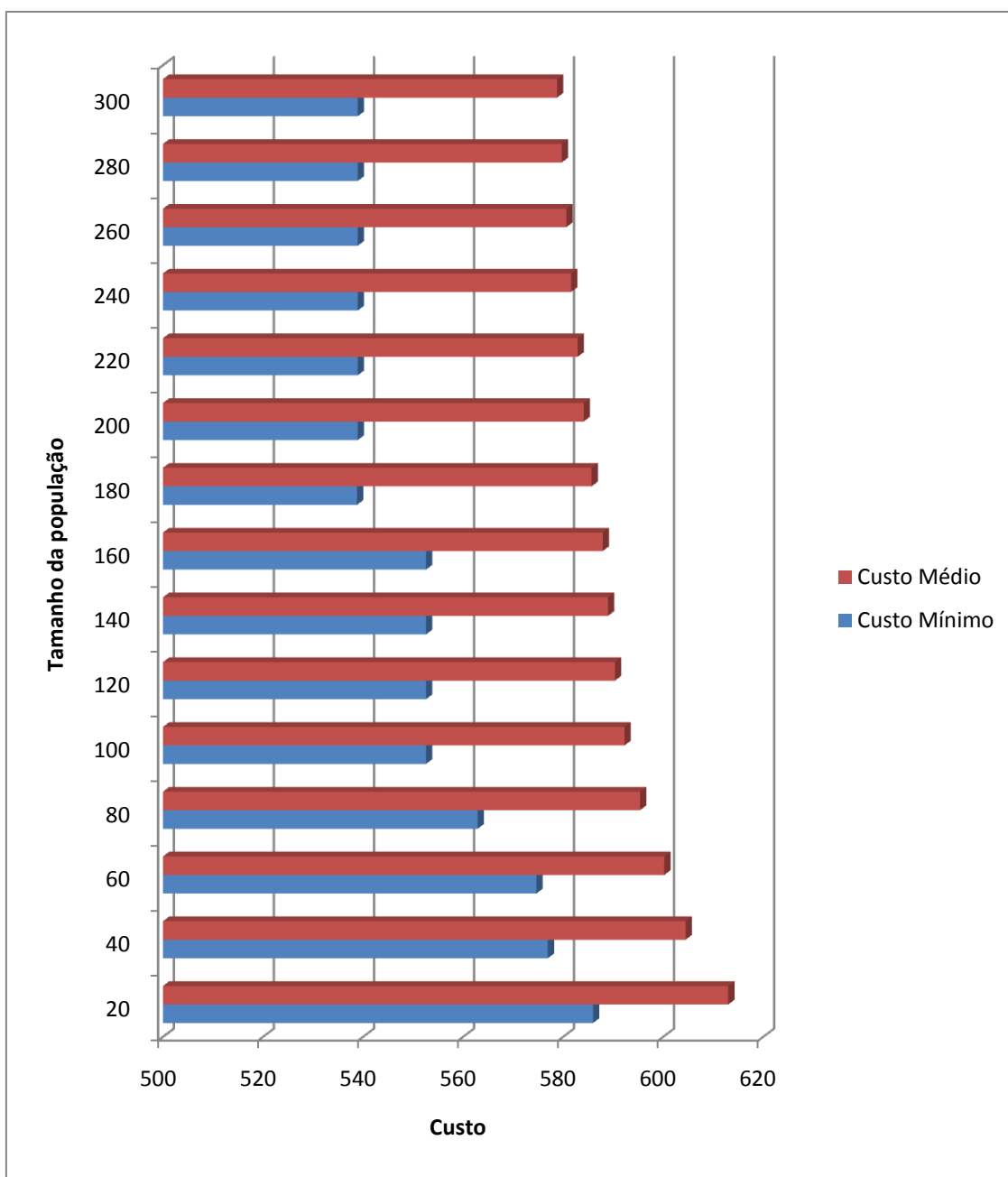


FIGURA 32 - GRÁFICO COMPARATIVO DE DIFERENTES TAMANHO DE POPULAÇÃO

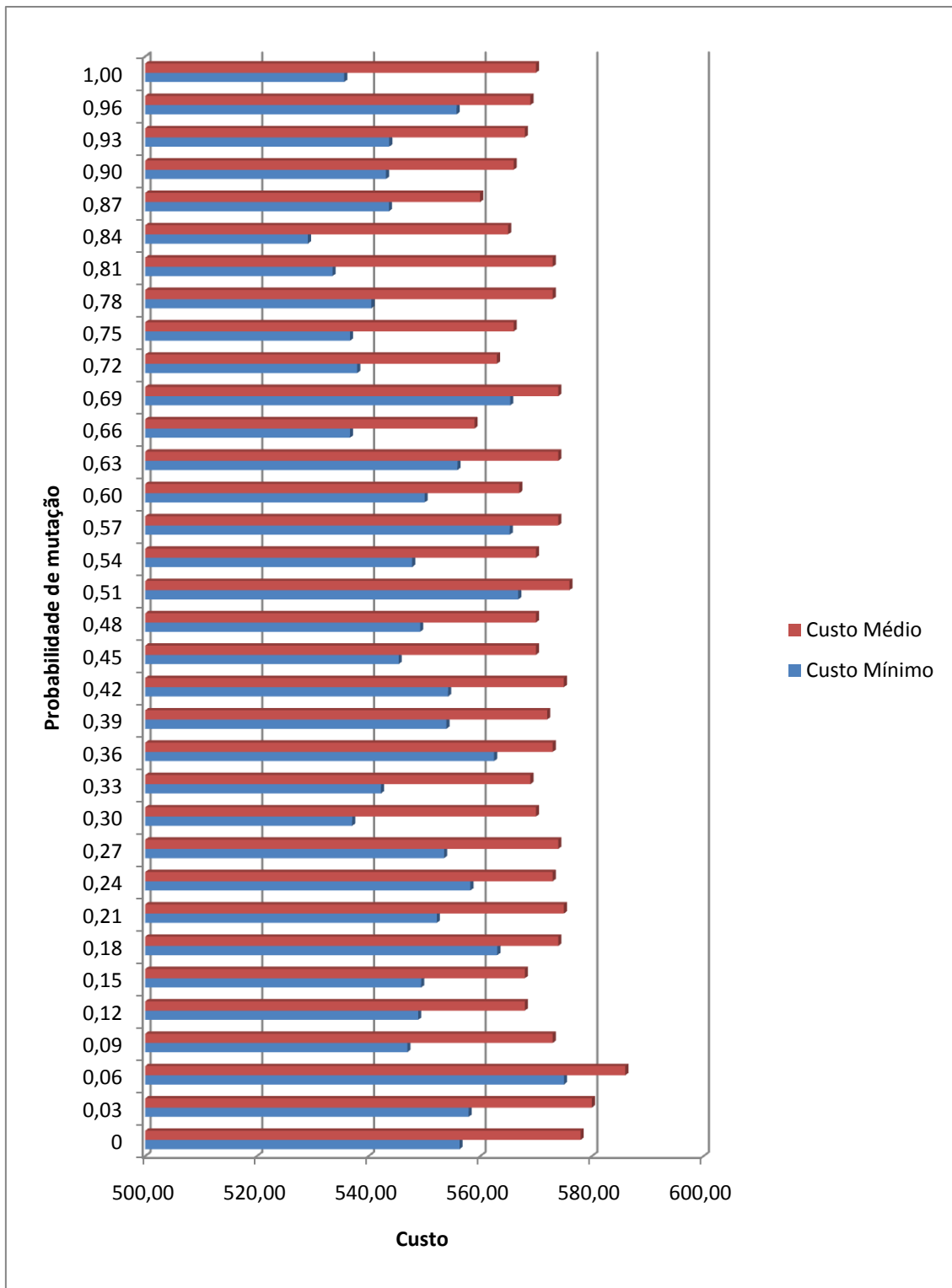


FIGURA 33 - GRÁFICO COMPARATIVO ENTRE DIFERENTES TAXAS DE MUTAÇÃO

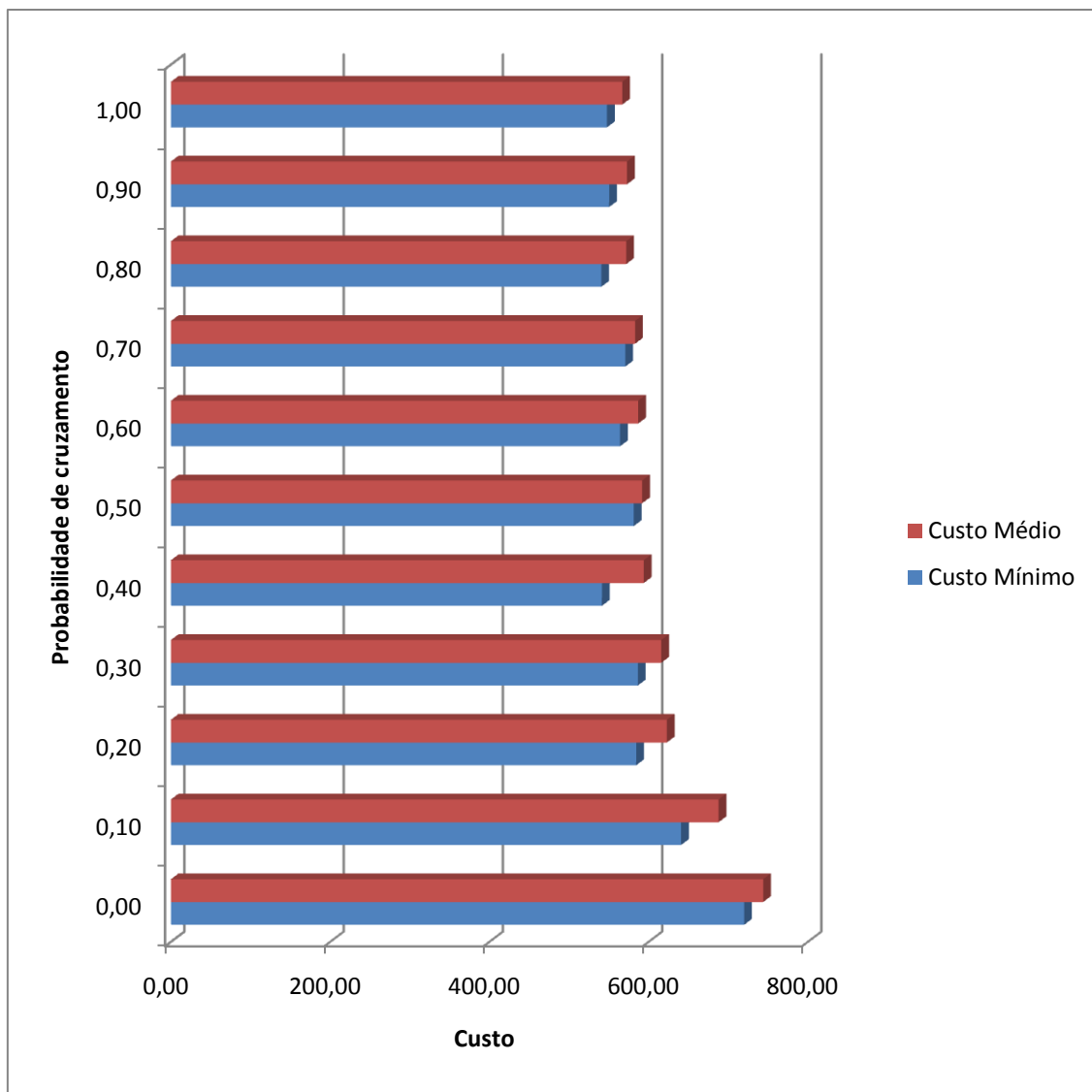


FIGURA 34 - GRÁFICO COMPARATIVO ENTRE DIFERENTES TAXAS DE CRUZAMENTO

5.3 Comparação com resultados encontrados na literatura

A tabela 5 apresenta uma comparação dos resultados obtidos utilizando o algoritmo genético com as melhores soluções conhecidas para várias instâncias do problema de roteamento de veículos, o nome de cada instância indica o número de pontos representados por n e o número de veículos necessários para a melhor solução conhecida representada por k (VRP Web, 2011).

TABELA 4 - COMPARAÇÃO ENTRE OS RESULTADOS OBTIDOS COM AS MELHORES SOLUÇÕES CONHECIDAS PARA VÁRIAS INSTÂNCIAS DO PROBLEMA DE ROTEIRIZAÇÃO DE VEÍCULOS

Instância	Melhor solução conhecida	Solução proposta				
	Custo Mínimo	Custo Mínimo	Custo Médio	Desvio Mínimo	Desvio Médio	Tempo Médio
E-n13-k4	290	278,43	278,64	-3,99%	-3,92%	4,6s
E-n23-k3	569	470,05	470,05	-17,39%	-17,39%	4,9s
E-n51-k5	521	527,67	558,75	1,28%	7,25%	12,1s
E-n76-k7	683	776,03	863,55	13,62%	26,43%	19,7s
E-n76-k8	735	835,12	954,08	13,62%	29,81%	20,7s

A tabela 5 mostra que o algoritmo proposto produz bons resultados para as instâncias dos problemas de roteamento testadas, com um tempo de resposta rápido e em alguns casos com soluções melhores que as conhecidas.

6 CONSIDERAÇÕES FINAIS

Este trabalho apresentou um módulo para roteamento de veículos utilizando um algoritmo genético para otimizar rotas de distribuição.

Este capítulo está dividido como segue a seção 6.1 apresenta as contribuições e conclusões e a 6.2 os trabalhos futuros.

6.1 Contribuições e conclusões

As contribuições deste trabalho são:

- a. Técnicas e algoritmos da área de otimização combinatória para resolução do PRV;
- b. Um algoritmo genético para o PRV para cálculo de rotas de distribuição respeitando a capacidade dos veículos e minimizando o custo total;
- c. Pesquisa e implementação de operadores genéticos para uma renovação eficaz da população;
- d. Uma representação gráfica para a apresentação dos resultados obtidos e simulação das rotas.

A partir destas contribuições, pode-se concluir que:

- a. É possível automatizar a construção de roteiros com custos otimizados levando em consideração a restrição da capacidade dos veículos;
- b. O PRV é um dos problemas mais estudados no problemas na área de otimização combinatória devido a sua aplicabilidade a diversos problemas no mundo real;
- c. Os algoritmos genéticos provaram ser um método de otimização combinatória eficaz no tratamento do PRV;

- d. Utilizando um algoritmo genético é possível obter soluções boas com custos computacionais relativamente baixos;

As seguintes observações foram obtidas:

- a) Os algoritmos genéticos podem ser utilizados para resolução de uma grande variedade de problemas de natureza combinatorial;
- b) Para uma melhor implementação do algoritmo genético é necessário escolher a melhor forma de representação da solução de acordo com o problema a ser resolvido;
- c) Os melhores resultados foram obtidos utilizando a seguinte combinação nos parâmetros: o método de seleção torneio, o operador de cruzamento OX e o operador de mutação ISM.

6.2 Trabalhos futuros

As contribuições alcançadas com este trabalho não encerram as pesquisas relacionadas a resolução do problema de veículos, mas abrem oportunidades para alguns trabalhos futuros:

- Utilizar um algoritmo para resolução de caminho mínimo em um sistema de informação geográfico (SIG) para obter os dados de distâncias de uma base de dados real;
- Realizar a integração do algoritmo de caminho mínimo com o algoritmo genético, para determinar as distâncias reais entre os depósitos;
- Pesquisar novos operadores genéticos, ou utilizar uma estratégia que combine diversos operadores genéticos, por exemplo, a aplicação de diferentes operadores de busca sempre que o processo evolutivo não conduzir a soluções de melhora;

- Pesquisar heurísticas para a formação da população inicial para reduzir o grau de aleatoriedade do algoritmo genético.

REFERÊNCIAS BIBLIOGRÁFICAS

CORMEN, T.; LEISERSON, C.; RIVEST, R.; STEIN, C. Algoritmos. 13ª ed. Rio de Janeiro: Rua Sete de Setembro, 2002. ISBN 85-352-0926-3.

ROSA, M. E. S. Um algoritmo heurístico para roteamento robusto e eficiente baseado em avaliação de fluxo máximo. 2008. Mestre em Informática – Universidade Federal do Paraná, 2008.

MALAQUIAS, N. G. L. Uso dos algoritmos genéticos para a otimização de rotas de distribuição. 2006. Mestre em Ciências – Universidade Federal de Uberlândia, 2006.

DAVIS Jr., C. A. Aumentando a eficiência da solução de problemas de caminho mínimo em SIG, 1997

PIMENTA, D. J. Algoritmo de Otimização para o Problema de Roteamento de Veículos no Transporte Conjunto de Cargas e de Passageiros, 2001

ALVARENGA, G. B. Um Algoritmo Híbrido para o Problema de Roteamento de Veículos Estático e Dinâmico com Janela de Tempo, 2005

DALBONI, F. L. Algoritmos Evolutivos Eficientes para um problema de roteamento de veículos, 2003

MONTANÉ, F. A. T. Um algoritmo de busca tabu adaptativo aplicado a problemas de roteamento de veículos com pedidos de coleta e entrega, 2006

GALVÃO, R. D.; BARROS, J. D.; FERREIRA, V. J. M.; HENRIQUES, H. B. S. Roteamento de veículos com base em sistemas de informação geográfica, 1997

HEINEN, M. R. Análise e Implementação de Algoritmos para o Roteamento de Veículos, 2005

RIBEIRO, G. M.; LORENA, L. A. N. Roteamento de veículos dinâmico usando algoritmos genéticos. XIX ANPET - Congresso de Pesquisa e Ensino em Transportes - Recife /PE - 7 a 11 de novembro de 2005

SOSA, N. G. M.; GALVÃO, R. D.; GANDELMAN, D. A. Algoritmo de busca dispersa aplicado ao problema clássico de roteamento de veículos, 2007

ROSSETTO, D. R.; GONZAGA, C. C. O algoritmo A* para busca de caminhos em grafos e o problema de alinhamento de proteínas, 2007

OLIVEIRA, H. C. B.; SOUZA, M. M.; ALVARENGA, G. B.; SILVA, R. M. A. Adaptação do Algoritmo Genético no Tratamento do Problema de Roteamento de Veículos com Janela de Tempo, 2004

GRAÇA, A. E. S. T. Novos algoritmos para problemas dinâmicos de roteirização de veículos com janela de tempo, 2009

LARRAÑAGA, P. et al. Genetic algorithms for the salesman problem: A review of representations and operators. Artificial Intelligence review, 1999

VRP Web, Disponível em: <http://neo.lcc.uma.es/radi-aeb/WebVRP/> Acesso em: 27 jun. 2011

CHRISTOFIDES, N., MINGOZZI, A. TOTH, P. Combinatorial Optimization, 1979