

CENTRO UNIVERSITÁRIO DO ESTADO DO PARÁ
ESCOLA DE NEGÓCIOS, TECNOLOGIA E INFORMAÇÃO
BACHARELADO EM ENGENHARIA DE COMPUTAÇÃO

ADILSON DE ALMEIDA NETO

**APLICAÇÃO DE UM ALGORITMO GENÉTICO PARA OTIMIZAÇÃO
DO ROTEAMENTO DE VEÍCULOS COLETIVOS EM REDE**

Belém

2018

CENTRO UNIVERSITÁRIO DO ESTADO DO PARÁ
ESCOLA DE NEGÓCIOS, TECNOLOGIA E INFORMAÇÃO
BACHARELADO EM ENGENHARIA DA COMPUTAÇÃO

ADILSON DE ALMEIDA NETO

**APLICAÇÃO DE UM ALGORITMO GENÉTICO PARA OTIMIZAÇÃO
DO ROTEAMENTO DE VEÍCULOS COLETIVOS EM REDE**

Trabalho de Curso na modalidade Monografia, apresentado como requisito parcial para obtenção do grau em Bacharelado em Engenharia de Computação do Centro Universitário do Estado do Pará – CESUPA, sob orientação do Professor Msc. Fábio dos Santos Ferreira

Belém

2018

Dados Internacionais de Catalogação-na-publicação (CIP)

Biblioteca do Cesupa, Belém – PA

Almeida Neto, Adilson de.

Aplicação de um algoritmo genético para otimização do roteamento de veículos coletivos em rede / Adilson de Almeida Neto; orientador Fábio dos Santos Ferreira. – 2018.

Trabalho de Conclusão de Curso (Graduação) – Centro Universitário do Estado do Pará, Bacharelado em Engenharia da Computação, Belém, 2018.

1. Computação evolutiva. 2. Otimização. 3. Algoritmos Genéticos. I. Ferreira, Fábio dos Santos. *orient.* II. Título.

ADILSON DE ALMEIDA NETO

**APLICAÇÃO DE UM ALGORITMO GENÉTICO PARA OTIMIZAÇÃO
DO ROTEAMENTO DE VEÍCULOS COLETIVOS EM REDE**

Trabalho de Curso na modalidade Monografia apresentado como requisito parcial para obtenção do grau em Bacharelado em Engenharia de Computação do Centro Universitário do Estado do Pará – CESUPA.

Data da Defesa: 15/06/2018

Banca Examinadora:

Prof. Orientador MSc. Fábio dos Santos Ferreira – CESUPA

Centro Universitário do Estado do Pará

Prof. Msc. Rodrigo Lisboa Pereira – UFRA

Universidade Federal Rural da Amazônia

Prof. Msc. Daniel Leal Souza – UFPA

Universidade Federal do Pará

Belém

2018

AGRADECIMENTOS

Agradeço primeiramente a minha família e amigos que me acompanham desde o começo do curso, agradeço ao meu orientador Fábio Ferreira, assim como aos Professores Rodrigo Lisbôa e Otávio Noura, que em conjunto com Marco Mollinetti, me orientaram oficialmente e não oficialmente durante minha iniciação científica.

Agradeço também a todos que contribuíram direta e indiretamente para este trabalho, amigos como Mario Serra, Tiago Morales, Rafael Gouveia, Filipe (Fyro), Flávio Ferreira, Cindy Teixeira, Lucas Pereira (Puga), entre tantos outros. Enfim, agradeço a todos que me acompanharam e ajudaram nesta etapa da minha vida.

RESUMO

Algoritmos genéticos são muito utilizados em várias classes de problemas de otimização, sobretudo em problemas combinatórios, onde o espaço de busca costuma ser demasiado grande para métodos exatos obterem soluções ótimas. Essa classe de problemas é de grande importância para modelos matemáticos que refletem situações do mundo real, como o roteamento de veículos. O seguinte trabalho objetiva utilizar um algoritmo genético para otimizar rotas de veículos coletivos ligados em rede, nessa configuração, as posições dos passageiros e dos motoristas são conhecidas em todos os momentos. Para que a otimização seja possível, o problema foi matematicamente modelado com base no *Capacitated Vehicle Routing Problem* (CVRP) com alterações para refletir a natureza do problema de roteamento de veículos coletivos ligados em rede. Além disso, esse trabalho compara a performance das rotas otimizadas do algoritmo genético com rotas não otimizadas, para observar o ganho obtido ao se utilizar essa técnica.

Palavras-chave: Computação Evolutiva. Otimização. Algoritmos Genéticos. Roteamento. Transporte. CVRP.

ABSTRACT

Genetic algorithms are used in a wide range of optimization problems, especially in combinatorial problems, where the search space is, in many occasions, too large for exact methods to achieve optimal solutions. This class of problems bear great importance for mathematical models that reflect real world situations, such as vehicle routing. The goal of this work is to utilize a genetic algorithm to optimize networked linked capacitated vehicles, in this configuration, all passengers and drivers positions are known in every moment. To make this optimization possible, the problem was mathematically modeled using the Capacitated Vehicle Routing Problem (CVRP) as inspiration with alterations reflecting the nature of the network linked capacitated vehicles. After that, this work also compares the obtained performance to a non-optimized route, so that the gain from using this method can be observed.

Keywords: Evolutionary Computing. Optimization. Genetic Algorithms. Routing. Transport. CVRP.

LISTA DE FIGURAS

Figura 1 – Representação de uma solução para o problema CVRP	16
Figura 2 – Representação de uma solução para o problema CVRP modificado	17
Figura 3 – Representação gráfica de um problema de minimização	19
Figura 4 – Diagrama de Atividades representando Algoritmo Genético	22
Figura 5 – Cromossomo de codificação binária.....	23
Figura 6 – Cromossomo de codificação octal.	23
Figura 7 – Cromossomo de codificação hexadecimal	23
Figura 8 – Cromossomo de codificação de permutação	24
Figura 9 – Cromossomo de codificação de valor	24
Figura 10 – Representação de uma roleta de seleção	26
Figura 11 – Escolha da seção de troca (OX)	28
Figura 12 – Limpeza no recebedor	28
Figura 13 – Alinhamento para troca.....	29
Figura 14 – Indivíduo final.....	29
Figura 15 – Escolha da seção de troca (MPX)	30
Figura 16 – Inversão no Indivíduo B	30
Figura 17 – Inserção da seção de A em B (MPX).....	31
Figura 18 – Mutação TWORS.....	32
Figura 19 – Mutação CIM	32
Figura 20 – Desvios padrão da distribuição 1	39
Figura 21 – Desvios padrão da distribuição 2	39
Figura 22 – Médias da distribuição 1	41
Figura 23 – Médias da distribuição 2	42
Figura 24 – Melhores da distribuição 1	42
Figura 25 – Melhores da distribuição 2	43
Figura 26 – Melhores do conjunto de pesos 1	44
Figura 27 – Melhores do conjunto de pesos 2	44

LISTA DE SIGLAS

GA	<i>Genetic Algorithm</i>
AG	<i>Algoritmo Genético</i>
TSP	<i>Travelling Salesman Problem</i>
CVRP	<i>Capacitated Vehicle Routing Problem</i>
CIM	<i>Centre Inverse Mutation</i>
OX	<i>Order Crossover</i>
MPX	<i>Multi Parent Crossover</i>

SUMÁRIO

1 INTRODUÇÃO	11
1.1 JUSTIFICATIVA.....	12
1.2 OBJETIVOS	12
1.2.1 Objetivo Geral	13
1.2.2 Objetivos Específicos	13
1.3 ESTRUTURA DO TRABALHO.....	13
2 TRANSPORTE E ROTEAMENTO	14
2.1 TRANSPORTE	14
2.2 ROTEAMENTO.....	15
2.2.1 Modelagem do CVRP	15
2.2.2 Modelagem do CVRP Modificado	16
2.2.3 Objetivo	17
2.3 ROTEAMENTO DE VEÍCULOS COLETIVOS EM REDE	18
3 OTIMIZAÇÃO E ALGORITMOS GENÉTICOS	19
3.1 OTIMIZAÇÃO	19
3.2 ALGORITMOS GENÉTICOS.....	20
3.3 CODIFICAÇÃO	22
3.3.1 Codificação Binária	23
3.3.2 Codificação Octal	23
3.3.3 Codificação Hexadecimal	23
3.3.4 Codificação de Permutação	24
3.3.5 Codificação de Valor	24
3.4 SELEÇÃO.....	24
3.4.1 Seleção Aleatória	25
3.4.2 Seleção por Roleta	25
3.4.3 Seleção por Torneio	26
3.5 CRUZAMENTO.....	27
3.5.1 OX	27
3.5.2 MPX	29

3.6 MUTAÇÃO	31
3.6.1 Mutação TWORS	32
3.6.2 Mutação CIM	32
3.7 CRITÉRIO DE PARADA.....	32
4 METODOLOGIA	34
4.1 CONSTRUÇÃO DO AG	34
4.1.1 Inicialização.....	34
4.1.2 Seleção	34
4.1.3 Cruzamento	35
4.1.4 Mutação	35
4.1.5 Critério de Parada.....	36
4.1.6 Fitness	36
4.2 ELABORAÇÃO DOS DADOS	37
4.3 EXPERIMENTOS	37
5 RESULTADOS	38
5.1 ANÁLISE DO ALGORITMO GENÉTICO	38
5.2 ANÁLISE DOS RESULTADOS	43
6 CONCLUSÃO	45
6.1 DIFICULDADES ENCONTRADAS.....	45
6.2 TRABALHOS FUTUROS	45
7 REFERÊNCIAS	47
APÊNDICE A – Conjunto de pesos 1	49
APÊNDICE B – Conjunto de pesos 2.....	50

1 INTRODUÇÃO

Não é possível se falar em grandes cidades sem mencionar o congestionamento, fator comum em quase todas. No Pará, cerca de 55 minutos são gastos por dia apenas com deslocamento, valor que contribui para uma perda entre 1.35% e 2.5% do PIB por município (VIANNA; YOUNG, 2015). Claramente qualquer avanço que diminua o tempo de espera e viagem contribui diretamente para a economia do país.

Em muitos países o trânsito causa danos de vários tipos: ao meio ambiente; à saúde pública e a produtividade de um país, nos Estados Unidos por exemplo se estima que em 2014 a cada um bilhão de dólares investido em transporte público existe um retorno em até 3.7 bilhões entre retornos de investimentos e ganhos de produtividade de modo geral (AMERICAN PUBLIC TRANSPORTATION ASSOCIATION, 2014).

Podemos definir trânsito como “conjunto de deslocamentos de pessoas e veículos nas vias públicas, dentro de um sistema convencional de normas, que tem, por fim, assegurar a integridade de seus participantes” (ROZESTRATEN, 1988, p.4). Grande parte dos problemas acontecem quando a fluidez do trânsito é interrompida, o excesso de veículos na rua causa poluição cujas consequências podem ser sentidas tanto na saúde pública como no meio ambiente, além dos travamentos que impedem que os trabalhadores cheguem nos seus locais de trabalho, sacrificando horas úteis de trabalho para milhares de pessoas (VIANNA; YOUNG, 2015).

Uma das soluções das cidades é a utilização de transporte público, que aumenta a quantidade de pessoas por veículo, aliviando os travamentos no trânsito. Porém, a oferta inadequada do transporte coletivo pode não somente prejudicar os que dependem diretamente dele como estimular o uso do transporte individual, aumentando a poluição e os congestionamentos criando um círculo vicioso (GOMIDE, 2006). O uso ampliado destes automóveis também favorece a dispersão das atividades na cidade, dificultando ainda mais o transporte para aqueles que dependem do transporte coletivo (GOMIDE, 2006).

Recentemente empresas como Uber e Lyft entraram no mercado de transporte particular usando um modelo similar ao dos taxistas, porém utilizando um aplicativo e certo controle inteligente dessa forma barateando o transporte individual, apesar

disso, em alguns casos o Uber em particular pode ter um efeito positivo no uso de transportes públicos, os resultados dependem muito do contexto da cidade antes da adoção do Uber (HALL, 2017), uma das possíveis explicações é que um motorista que seguiria sozinho pode em alguns casos compartilhar a corrida do Uber.

Nesse contexto, um problema a ser analisado é o da eficiência do transporte público: por muitos anos o transporte público se deu da mesma forma, seja por ônibus ou metrô, utilizando paradas e rotas fixas para transportar passageiros do ponto A ao ponto B, deve-se analisar também outras formas de realizar esse transporte, como a inserção de inteligência nas rotas. O objetivo deste trabalho é o de aplicar um algoritmo genético ao problema de roteamento no contexto do transporte coletivo e analisar os resultados. Essa otimização é válida pois ocorre sem custos com investimento pesado em infraestrutura ou equipamento, apenas o custo necessário para otimizar as rotas dos veículos.

1.1 JUSTIFICATIVA

Em paralelo aos problemas causados pelo trânsito e pela ineficiência do transporte urbano estão os algoritmos de roteamento, que já vem sendo estudados a muito tempo e aplicados nas mais diversas áreas, a aplicação aqui discutida seria a criação de rotas para veículos de transporte coletivo em rede, estudos anteriores exploraram aplicações semelhantes, como a redistribuição de bicicletas em um ambiente urbano (LIN; CHOU, 2012). A viabilidade dessa aplicação se dá devido ao avanço dos algoritmos dedicados a solucionar este tipo de problema de roteamento, como o *Capacitated Vehicle Routing Problem (CVRP)* e *Travelling Salesman Problem (TSP)* (LAPORTE, 1992).

Assim como os investimentos em infraestrutura geram frutos positivos ao transporte, também se espera que investimentos em eficiência diminuam os congestionamentos nas grandes cidades, contribuindo assim para a vida do cidadão.

1.2 OBJETIVOS

Os objetivos desta pesquisa são divididos em geral e específicos, de acordo com os tópicos a seguir.

1.2.1 Objetivo Geral

O objetivo deste trabalho é o de aplicar um algoritmo genético ao problema de roteamento no contexto do transporte coletivo e analisar os resultados.

1.2.2 Objetivos Específicos

- Expor os conceitos relacionados ao Algoritmo Genético.
- Expor noções básicas de transporte e do problema CVRP.
- Implementar um Algoritmo Genético para resolver o problema CVRP.
- Analisar os resultados estatísticos obtidos.

1.3 ESTRUTURA DO TRABALHO

Além deste capítulo introdutório, o trabalho está organizado da seguinte maneira:

- Capítulo 2 – Transporte e Roteamento
- Capítulo 3 – Otimização e Algoritmos Genéticos
- Capítulo 4 – Metodologia
- Capítulo 5 – Resultados
- Capítulo 6 – Conclusão

Onde o capítulo 2 tratará do conceito de transporte no ambiente urbano e o problema de roteamento da computação e como se relacionam estes tópicos, o capítulo 3 explanará a otimização e o algoritmo genético como solução para um problema otimizável, o quarto capítulo explica como foram feitos os testes e obtidos os resultados, assim como detalha a geração dos dados utilizados nos testes, no capítulo 5 se analisam os resultados e a performance do algoritmo genético. Por fim, no capítulo 6 é feita a conclusão do trabalho e as perspectivas futuras.

2 TRANSPORTE E ROTEAMENTO

Esta seção tratará tanto do transporte como conceito para o ambiente urbano como do roteamento como problema da computação. O objetivo é esclarecer como a computação se aplica e beneficia o transporte.

2.1 TRANSPORTE

O transporte é uma característica fundamental às cidades e é definido pelas relações dos indivíduos com o espaço. Por um grande período as relações de transporte nas grandes cidades eram marcadas por jornadas simples, onde cada trabalhador ou passante ia de sua casa em direção ao centro. Atualmente esse cenário vem sendo substituído por uma suburbanização, onde as pessoas vêm se distanciando das cidades em função dos altos custos associados a viver perto do centro e até mesmo nas periferias da cidade, assim o transporte muda de uma movimentação próxima para um deslocamento cada vez maior e cada vez mais caracterizado pelo transporte individual (BANISTER, 1995).

Evidências apontam que investimentos em novas estradas e infraestruturas de transporte tendem a aumentar o tráfego em economias avançadas piorando os problemas de mobilidade. Esse deslocamento do transporte para longe dos centros das cidades tem aumentado o uso do transporte particular e afetado diretamente na mobilidade daqueles que não tem veículo próprio (BANISTER, 1995).

Uma das soluções das cidades para aliviar o congestionamento e melhorar o transporte é priorizar e aumentar a eficiência do transporte público, um meio de transporte de caráter coletivo, que pode realizar o transporte de dezenas de pessoas sem criar um tráfego da mesma proporção (BANISTER, 1995), existem campos da ciência que estudam a otimização do transporte público no mérito da urbanização e organização das cidades porém neste trabalho foi explorado o roteamento, mais especificamente o *Capacitated Vehicle Routing Problem (CVRP)*, problema que pode ser traduzido como Problema de Roteamento de Veículo Capacitado.

2.2 ROTEAMENTO

Este item trata do roteamento como problema da computação, expondo sua definição e aplicabilidade para o problema real de transporte.

2.2.1 Modelagem do CVRP

Existem diversas formas de modelar matematicamente um problema de roteamento. O modelo usado nesse trabalho é uma versão modificada do CVRP.

De acordo com Laporte (1987), o CVRP pode ser definido como um grafo:

$$G = (V, A) \quad (1)$$

Onde:

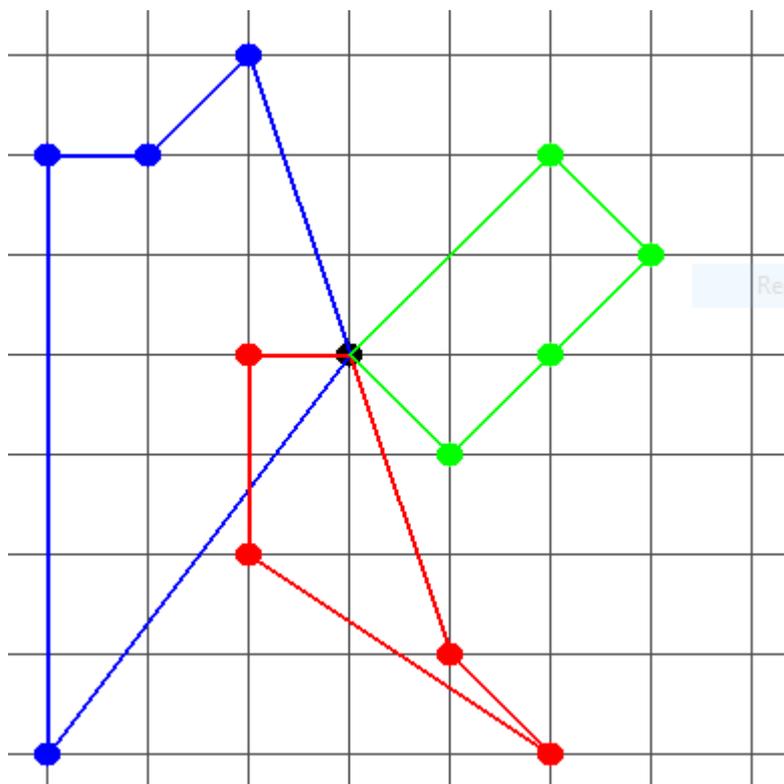
- V são os vértices
- A são as arestas

Cada aresta é associada a uma função de custo denominada C, além disso existem m veículos no vértice 1, que é considerado a origem. Os veículos saem da origem em direção aos pontos de distribuição com uma carga C que será distribuída nos pontos de parada. Uma solução para o problema clássico consiste em uma rota onde:

- i. Cada vértice é visitado apenas uma vez por um único veículo.
- ii. Todos os veículos iniciam sua rota e terminam na origem.
- iii. Todas as restrições são satisfeitas.

O CVRP com uma solução é ilustrado na figura 1, onde é possível observar a origem representada por um ponto preto e as arestas e vértices são representadas por linhas e círculos respectivamente, sendo esses da cor azul, verde ou vermelho, é comum um gráfico de CVRP conter arestas e vértices de cores diferentes para simbolizar diferentes veículos de uma solução.

Figura 1 – Representação de uma solução para o problema CVRP



Fonte: Autor (2018)

2.2.2 Modelagem do CVRP Modificado

O CVRP modificado proposto como modelagem para o problema tratado neste trabalho segue a definição da fórmula 1, sendo um grafo não direcionado constituído por vértices e arestas, onde existe uma função de custo associada a cada aresta, representando a distância entre os pontos. Algumas modificações se fazem necessárias para adaptar o modelo:

- i. Cada vértice tem um dois valores não negativos associados, o primeiro sendo a quantidade de passageiros no ponto e o segundo a quantidade de passageiros a ser adicionado a esse total a cada iteração, denominados respectivamente total de passageiros e peso do vértice.
- ii. Os veículos saem de um ponto escolhido, que será denominado a origem, porém não se faz nenhum tipo de requisito adicional sobre este ponto, além de pertencer ao conjunto de vértices.

iii. Os veículos quando num vértice devem coletar passageiros diminuindo o total de passageiros no vértice pela mesma quantidade que será adicionada à carga total do veículo.

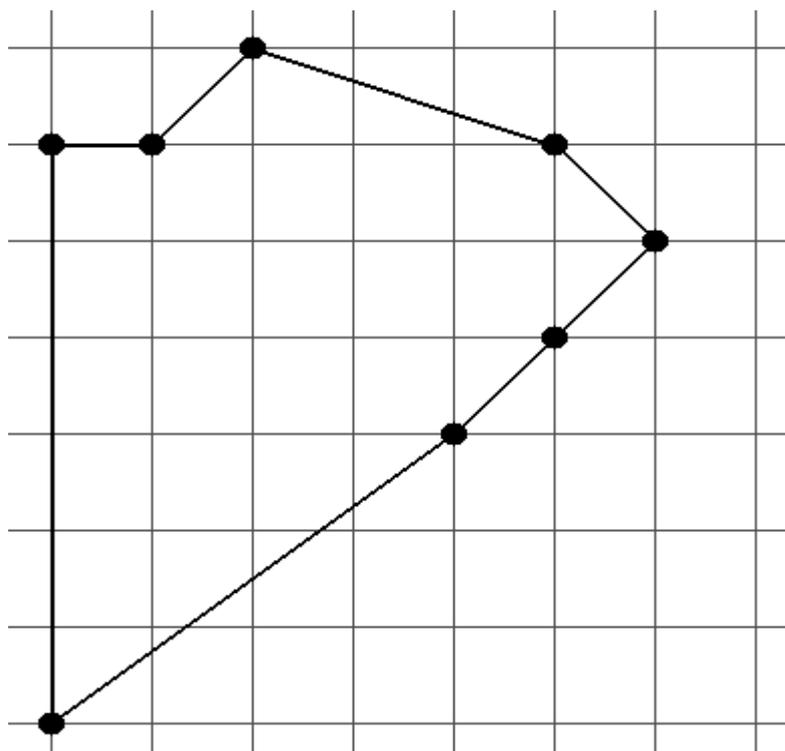
A solução para este problema se trata de uma rota onde:

i. Todos os passageiros são coletados, eventualmente.

ii. Nenhum veículo excede sua carga máxima de N passageiros em nenhum momento.

Na figura 2, podemos observar uma solução para a versão modificada do CVRP, onde os pontos pretos representam os vértices e as linhas pretas as arestas, cada um destes vértices podendo conter 0 ou mais passageiros, a solução nesse caso se trata da ordem pela qual esses pontos são percorridos.

Figura 2 – Representação de uma solução para o problema CVRP modificado



Fonte: Autor (2018)

2.2.3 Objetivo

O objetivo desta otimização é diminuir o tempo médio de viagem e espera dos usuários do transporte coletivo, sendo ele público ou privado, já que o algoritmo pode ser usado independente do modelo de funcionamento do transporte.

2.3 ROTEAMENTO DE VEÍCULOS COLETIVOS EM REDE

O modelo padrão de transporte público é constituído por uma frota de ônibus que seguem rotas pré-definidas com paradas obrigatórias por onde os passageiros podem entrar nos ônibus, a modificação a esse modelo seriam duas:

i. Os passageiros e os motoristas estariam ligados em rede, onde cada um teria conhecimento da posição de todos os outros.

ii. As paradas obrigatórias seriam fixas, porém a rota que as liga seria dinâmica, já que as paradas poderiam mudar de ordem.

Se espera que a flexibilidade alcançada com essa modificação possa aumentar a eficiência desse tipo de transporte por otimizar os tempos de viagem e de espera, além de que essas modificações são necessárias para que a modelagem matemática do modelo CVRP modificado seja aplicável ao problema.

3 OTIMIZAÇÃO E ALGORITMOS GENÉTICOS

Esta seção explicará o conceito de otimização, além de apresentar o algoritmo genético e seus conceitos associados.

3.1 OTIMIZAÇÃO

Formalmente, a otimização matemática é o processo de formulação e solução de um problema restringido com a forma matemática:

Minimizar a função objetivo

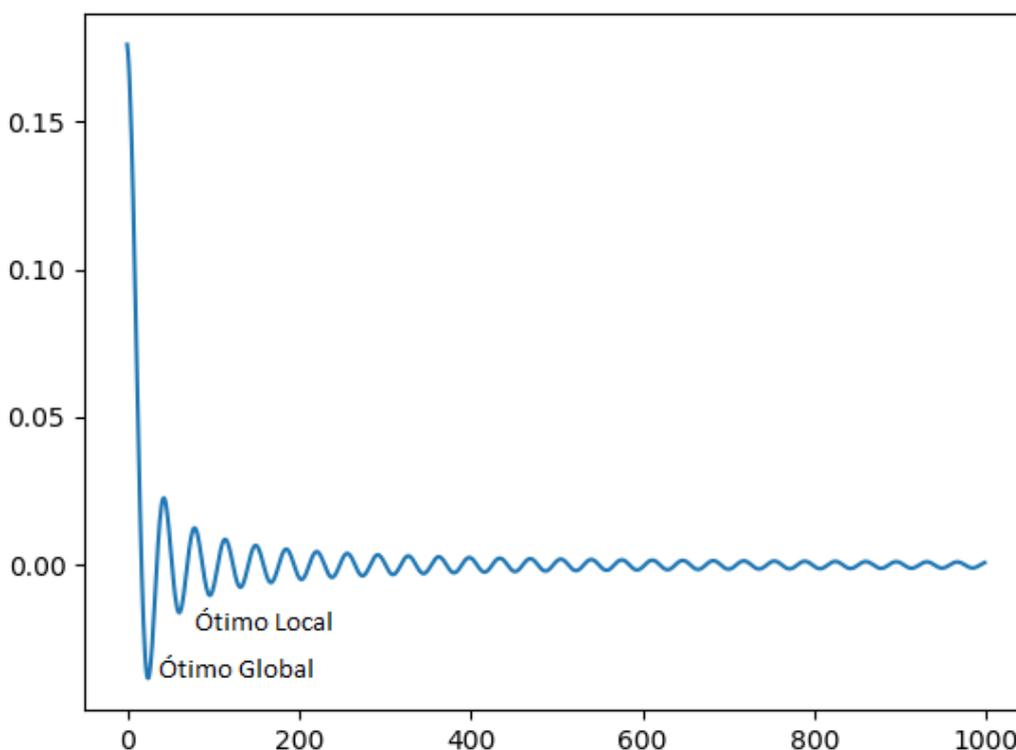
$$f(x), x = [x_1, x_2, \dots, x_n]^T \in R \quad (2)$$

Sujeito as restrições (SNYMAN, 2005):

$$g_j \leq 0, j = 1, 2, 3, \dots, m \quad (3)$$

$$h_j \leq 0, j = 1, 2, 3, \dots, n \quad (4)$$

Figura 3 – Representação gráfica de um problema de minimização



Fonte: Autor (2018)

A figura 3 demonstra a representação gráfica de um problema de minimização, é possível observar no gráfico dois pontos, o ótimo global e ótimo local, o primeiro é a solução desejada e o segundo é ponto no qual o algoritmo pode ficar preso, por se tratar de um pico ou vale na sua vizinhança.

Nesse problema, o objetivo é formalizar o modelo matematicamente, para que se possa computar a precisão de uma solução somente com operações matemáticas.

3.2 ALGORITMOS GENÉTICOS

Durante o período de 1950 – 1960, muitos cientistas da computação desenvolveram independentemente sistemas evolucionários de computação com a esperança de que a evolução poderia ser utilizada como molde para criar algoritmos de otimização, a ideia nesses algoritmos era de que existiriam certas soluções candidatas que sofreriam alterações com operadores inspirados pela seleção natural e variabilidade genética (MITCHELL, 1998).

Apesar de se basearem no mesmo fenômeno natural, alguns diferentes algoritmos surgiram durante esse período citado, entre eles podemos destacar:

- i. Algoritmos Genéticos.
- ii. Programação Genética.
- iii. Estratégias Evolutivas.
- iv. Programação Evolutiva.

A diferença essencial entre esses diferentes algoritmos vem da forma de representação genética e na diferença nos operadores, tanto de reprodução quanto de seleção (SIVANANDAM; DEEPA, 2007).

O algoritmo genético é portanto um algoritmo que adapta processos naturais de seleção e mutação para propósitos de otimização matemática, inventando por John Holland (HOLLAND, 1992). Uma das peculiaridades do Algoritmo Genético (AG) é que diferentemente dos outros métodos computacionais bioinspirados, ele não foi criado com o intuito de resolver problemas em particular, mas sim o de simular os processos de seleção natural e formalizar o estudo deste processo e desenvolver formas de adaptar esse tipo de processo para a solução de problemas reais (MITCHELL, 1998).

O apelo da evolução para estes cientistas era que muitos problemas reais precisam de algoritmos que sejam adaptativos, que possam apresentar bom desempenho em ambientes em mudança e complexos. Outra motivação é a de que a inteligência exigida para certos problemas pode ser elevada demais para uma abordagem “*top-down*”, onde todas as regras do sistema são especificadas explicitamente no programa. O algoritmo genético se encaixa numa abordagem “*bottom-up*” onde regras simples com interação massiva em paralelo podem levar a comportamentos complexos (MITCHELL, 1998).

Dado um problema matematicamente definido e uma representação adequada, um algoritmo genético pode ser definido pelas etapas do fluxograma mostrado na figura 4. Onde se deve fazer algumas observações:

i. Na inicialização, várias soluções viáveis para o problema em questão devem ser geradas, chamadas de indivíduos ou soluções candidatas, a representação dessa solução é o cromossomo, os genes de um indivíduo.

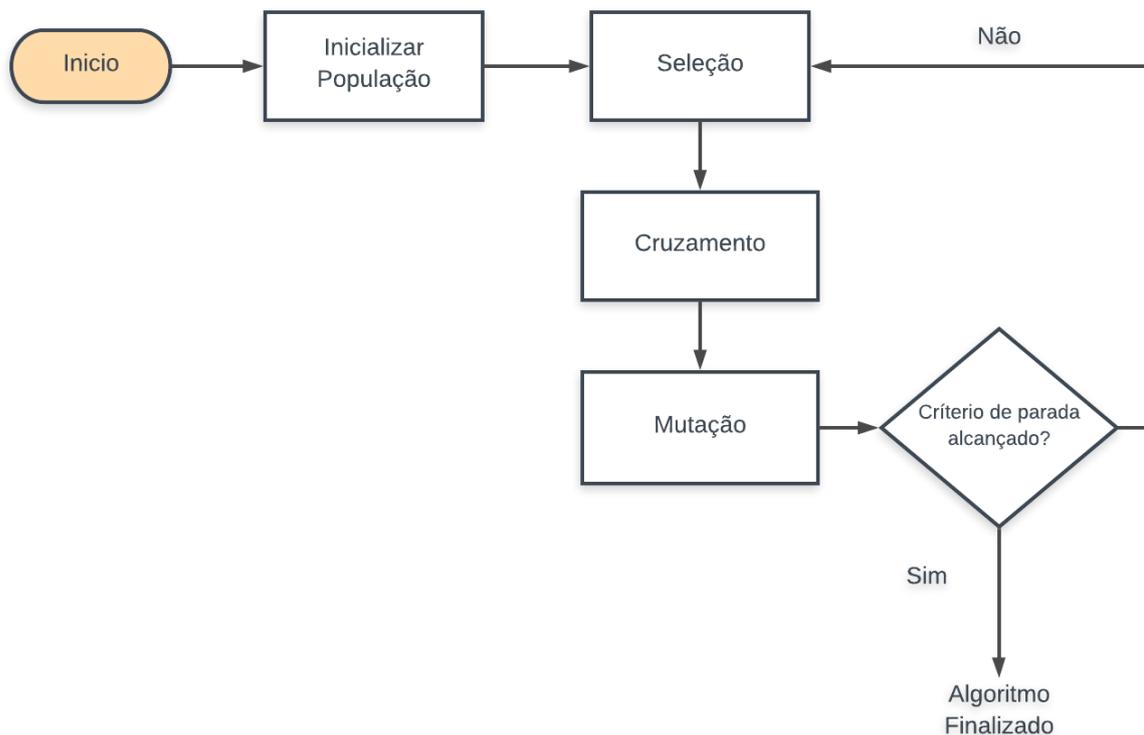
ii. Na seleção um indivíduo é selecionado baseado em um peso associado a ele, conhecido como *fitness*, calculado por via de uma função que depende da modelagem do problema.

iii. Na fase de cruzamento, indivíduos são selecionados dois a dois para criar uma população filha, esses pares de indivíduos são misturados utilizando alguma função apropriada ao problema com o fim de gerar um par de indivíduos filhos.

iv. Na fase de mutação as soluções candidatas serão modificadas levemente, para introduzir uma certa aleatoriedade na população, a mutação também ocorre por meio de uma função que irá ser modelada de acordo com o problema.

v. O critério de parada comumente se dá de duas formas: um limite máximo de execução para o algoritmo; uma qualidade mínima da solução, para que esta seja aceita.

Figura 4 – Diagrama de Atividades representando Algoritmo Genético



Fonte: Autor (2018)

Esse processo todo se repete até que o tempo limite tenha sido esgotado ou uma solução satisfatória foi encontrada (MITCHELL, 1998).

3.3 CODIFICAÇÃO

O algoritmo genético tenta resolver problemas baseando-se na seleção natural. Para que isso ocorra o código genético de cada indivíduo da população precisa representar de alguma forma o problema em questão, sendo que o processo pelo qual cada gene ou alelo é representado é chamado de codificação (SIVANANDAM; DEEPA, 2007). Denominamos o conjunto de todos os genes de um indivíduo de genótipo, genes ou cromossomo.

A seguir serão apresentadas algumas das representações comuns em algoritmos genéticos nas próximas seções.

3.3.1 Codificação Binária

Na codificação binária, o cromossomo é representado como um texto binário, composto de uns e zeros, a interpretação desse texto binário varia de acordo com o problema e essa é uma codificação clássica para o AG (SIVANANDAM; DEEPA, 2007), na figura 5 podemos observar um exemplo de codificação binária.

Figura 5 – Cromossomo de codificação binária



Fonte: Autor (2018)

3.3.2 Codificação Octal

Segue a lógica da codificação binária, o cromossomo é representado por um texto, porém este texto é composto por números de 0 até 7, sendo uma notação numérica octal (SIVANANDAM; DEEPA, 2007), na figura 6 podemos observar um exemplo de codificação octal.

Figura 6 – Cromossomo de codificação octal.



Fonte: Autor (2018)

3.3.3 Codificação Hexadecimal

Nesta codificação o cromossomo é representado como um texto composto por números na base hexadecimal, ou seja, números de 0 até 9 e letras de A até F na ordem alfabética (SIVANANDAM; DEEPA, 2007), a figura 7 exemplifica um cromossomo com codificação hexadecimal.

Figura 7 – Cromossomo de codificação hexadecimal



Fonte: Autor (2018)

3.3.4 Codificação de Permutação

Nesta codificação o cromossomo é composto por um conjunto de números representando uma ordem ou caminho a se seguir (SIVANANDAM; DEEPA, 2007). Esta é a codificação utilizada neste trabalho, por se tratar da ordem de pontos que um veículo seguirá na sua rota.

Na figura 8 podemos ver um exemplo de codificação de permutação, qualquer indivíduo na população deve ter o mesmo conjunto de valores, porém em ordens diferentes.

Figura 8 – Cromossomo de codificação de permutação



Fonte: Autor (2018)

3.3.5 Codificação de Valor

Nesta codificação o cromossomo é composto por um conjunto de valores podendo estes serem numéricos ou textuais, este tipo de codificação pode ser especialmente útil para certos tipos de problema apesar de normalmente ser necessário desenvolver operadores de mutação e cruzamento especializados (SIVANANDAM; DEEPA, 2007), um exemplo da codificação de valor é ilustrado na figura 9.

Figura 9 – Cromossomo de codificação de valor



Fonte: Autor (2018)

3.4 SELEÇÃO

A seleção é a fase do algoritmo genético (AG) na qual a população atual será usada para criar uma nova população, com características e *fitness* semelhantes aos pais. Antes desta fase ocorrer, no entanto, é preciso que se tenha uma medida da qualidade de cada indivíduo, a medida da qualidade do cromossomo do indivíduo é

denominada *fitness* e é computada com uma função desenvolvida para o problema (MCCALL, 2005).

A partir do momento que existe uma medida de *fitness* para cada elemento da população, este valor será usado como discriminante para a seleção. A ideia é simular uma certa pressão seletiva, onde soluções melhores têm mais chances de se reproduzir (MCCALL, 2005).

O objetivo desta pressão seletiva é o de gerar indivíduos com *fitness* próximo ou até superior aos pais, indicando novos caminhos de busca para o algoritmo. A intensidade com a qual essa seleção se baseia no *fitness* afetará diretamente a população filha. Uma seleção muito intensa deixará a população somente com os melhores, desta forma reduzindo a variabilidade genética e a chance para progresso, enquanto que uma seleção muito fraca atrasará demais a evolução da população (MITCHELL, 1998). Algumas formas de realizar essa seleção a partir do *fitness* serão demonstradas nas próximas seções.

3.4.1 Seleção Aleatória

Nessa técnica as soluções são selecionadas aleatoriamente da população para o cruzamento (SIVANANDAM; DEEPA, 2007), essa é a técnica mais simples, porém não reflete em nada a função *fitness* e, portanto, o AG se comporta de forma semelhante a uma busca cega.

3.4.2 Seleção por Roleta

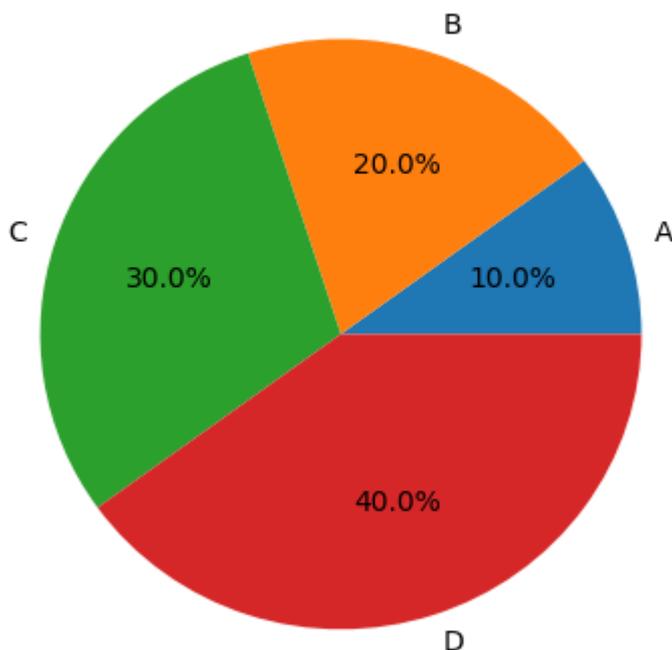
A seleção de roleta é uma das técnicas tradicionais de seleção para Algoritmos Genéticos, se trata de escolher as soluções para cruzamento com uma probabilidade baseada no seu *fitness*, funciona da seguinte forma (SIVANANDAM, DEEPA, 2007):

- i. É calculada a soma do *fitness* de todas as soluções
- ii. A probabilidade de cada solução ser selecionada é proporcional à sua própria *fitness* dividida pela soma calculada no primeiro passo.

O nome seleção por roleta vem do fato de que é possível representar essa seleção como uma roleta sendo girada, onde cada fatia dessa roleta é proporcional

ao *fitness* de cada indivíduo como na figura 10, onde a proporção dos indivíduos A, B, C e D são representadas em porcentagem.

Figura 10 – Representação de uma roleta de seleção



Fonte: Autor (2018)

3.4.3 Seleção por Torneio

A seleção por torneio se destaca por proporcionar a possibilidade de ajustar a pressão seletiva e a diversidade da população utilizando os parâmetros do torneio. Ela funciona da seguinte forma (SIVANANDAM; DEEPA, 2007):

- i. O torneio se inicia selecionando N indivíduos da população aleatoriamente de forma uniforme, esse número N é denominado o tamanho do torneio.
- ii. O indivíduo com o maior *fitness* entre esses N indivíduos é selecionado.

Esses dois passos se repetem até que se tenham indivíduos suficientes para preencher a nova população.

A seleção por torneio será a utilizada neste trabalho, por sua flexibilidade e simplicidade de implementação.

3.5 CRUZAMENTO

Pode se dizer que o cruzamento é um dos principais fatores do algoritmo genético (MITCHELL, 1998), pois se trata do processo pelo qual os cromossomos selecionados na fase da seleção são recombinados ou misturados para formar novos indivíduos, o objetivo é simular a mistura de material genético que ocorre quando dois organismos se reproduzem com o fim de inserir variabilidade na população (MCCALL, 2005). Claramente existe mais de uma interpretação para função do cruzamento no algoritmo genético, com alguns autores, seu uso se dá como uma forma de “macromutação”, possibilitando grandes saltos pelo espaço de busca (MITCHELL, 1998).

O operador de cruzamento está intimamente ligado à codificação e o *fitness*, seu sucesso ou fracasso dependerá em grande parte da sintonia com essas duas partes do algoritmo genético (MITCHELL, 1998). Por este motivo, nas próximas seções se exemplificarão os cruzamentos compatíveis com a codificação escolhida para o problema tema do trabalho (codificação de permutação).

3.5.1 OX

O cruzamento OX (Ordered Crossover) Tem seu nome por combinar a ordem dos elementos de duas soluções para gerar uma, de acordo com Moscato (1989), tem como propósito misturar duas rotas com esse processo, utilizando um certo reordenamento. Para isso acontecer, primeiramente denominamos o primeiro e segundo indivíduo de A e B respectivamente, e seguimos os seguintes passos:

- i. Selecionar uma seção entre o começo e o fim de ambos os cromossomos A e B como ilustrado na figura 11.
- ii. Para cada elemento desta seção em A, substituir os valores em B por asteriscos ou qualquer símbolo de escolha, a figura 12 mostra esse processo.
- iii. Mover estes asteriscos para a seção de troca, mantendo a ordem dos alelos em B, como na figura 13.

iv. Obter o indivíduo final substituindo os asteriscos em B pela seção de troca em A, como na figura 14.

Figura 11 – Escolha da seção de troca (OX)



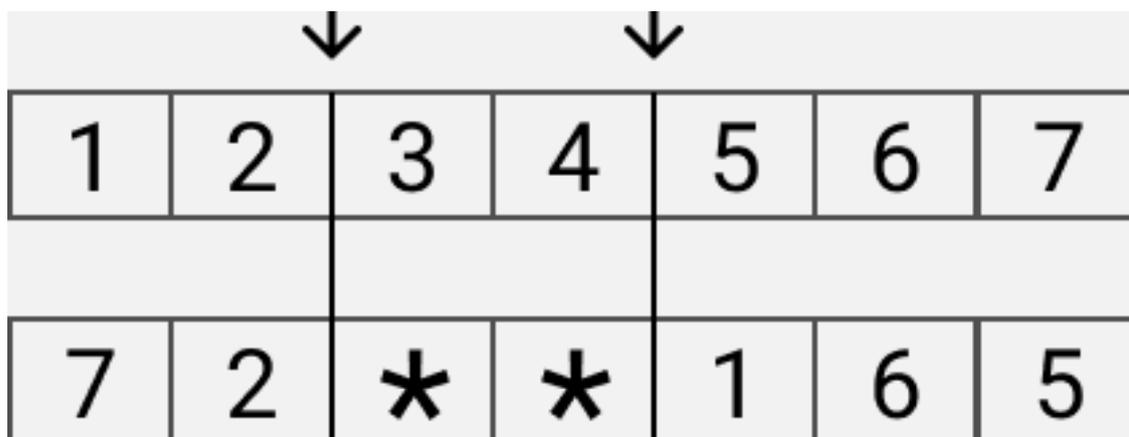
Fonte: Autor (2018)

Figura 12 – Limpeza no recebedor



Fonte: Autor (2018)

Figura 13 – Alinhamento para troca



Fonte: Autor (2018)

Figura 14 – Indivíduo final



Fonte: Autor (2018)

3.5.2 MPX

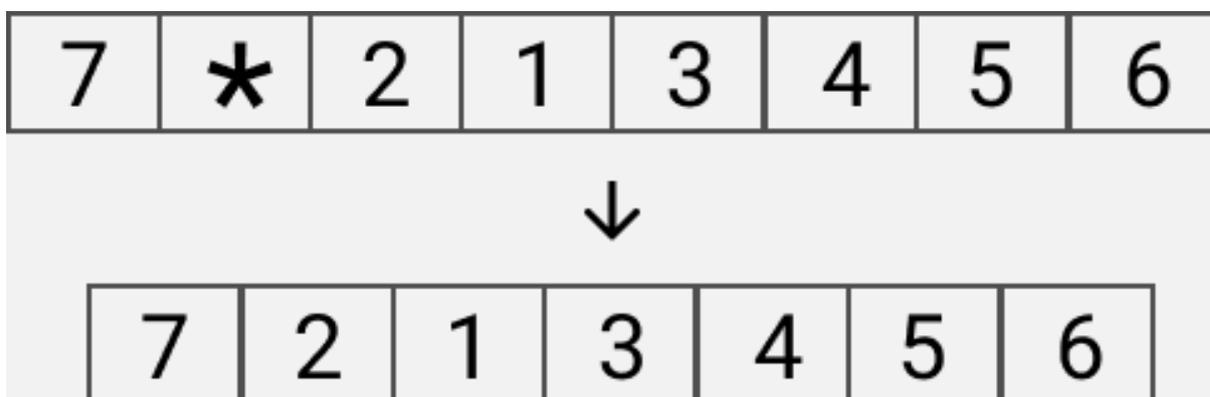
Segundo (MOSCATO, 1989), o cruzamento MPX ou *Multi Parent Crossover* segue a mesma lógica de doação do OX, porém com algumas modificações. Temos novamente os cromossomos A e B e seguimos os seguintes passos:

i. Se seleciona uma seção correspondente em ambos A e B para ser denominada a seção de troca, esta fase está representada na figura 15.

ii. Ocorre uma inversão em alguma parte do indivíduo, de forma que em B, o primeiro e o último elemento da seção de troca em A estejam lado a lado, esta fase está representada na figura 16, onde ocorre uma inversão em 65 para que o 3 fique ao lado do 5.

iii. Nesta última fase, a seção de troca de A é inserida entre o seu primeiro e último elemento em B, que foram devidamente alinhados no passo ii, construindo um cromossomo com elementos repetidos, os elementos repetidos fora da seção de troca são removidos, essa operação é ilustrada na imagem 17.

Figura 17 – Inserção da seção de A em B (MPX)



Fonte: Autor (2018)

3.6 MUTAÇÃO

Após o cruzamento, os cromossomos são sujeitos à mutação, nesta fase cada indivíduo sofrerá uma mutação com uma probabilidade denominada *mutation rate* ou taxa de mutação. Essa taxa proporciona um certo controle sobre a troca de exploração por especialização, exploração no sentido de explorar o espaço de busca e especialização no sentido da otimização fina de uma solução em particular.

Como foi visto na seção de otimização, além do ótimo global, a melhor solução do espaço de busca, existe uma série de ótimos locais que podem fazer com que o algoritmo genético se prenda a uma região do espaço de busca, a mutação resolve esse problema ao gerar perturbações no código genético de indivíduos da população, aumentando a variabilidade e proporcionando à população uma possibilidade de escapar de ótimos locais (ABDOUN, 2012).

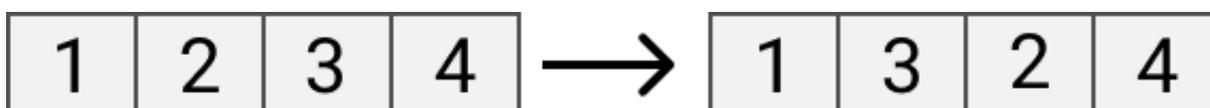
Essa visão de que a mutação é responsável por impedir ótimos locais é comum na comunidade científica desde o livro *Adaptation in Natural and Artificial Systems* de Holland, esse papel, visto muitas vezes como secundário é incomum em outros métodos como programação evolucionária, onde a mutação é a única forma de variabilidade na população (nas formas clássicas da programação evolucionária). Recentemente, no entanto, a comunidade científica vem dando cada vez mais atenção à mutação como um método de inserir variabilidade na população (MITCHELL, 1998).

O operador de mutação opera sobre a codificação, de forma que a solução modificada seja uma solução válida para o problema e siga a codificação estabelecida, para o problema CVRP com codificação de permutação, algumas formas de mutação são apropriadas e serão expostas nas próximas seções.

3.6.1 Mutação TWORS

Nesse tipo de mutação dois genes aleatórios de um cromossomo são trocados de posição (ABDOUN, 2012), como ilustrado na figura 18.

Figura 18 – Mutação TWORS

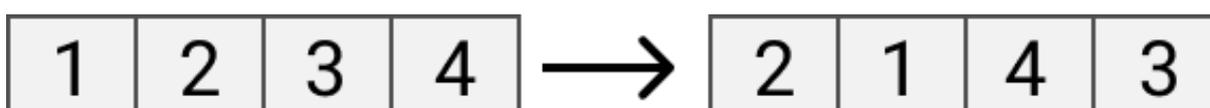


Fonte: Autor (2018)

3.6.2 Mutação CIM

A mutação CIM, também pode ser denominada *Centre Inverse Mutation* ou mutação de inversão de centro se trata de dividir o cromossomo em duas regiões e gerar um novo indivíduo com as duas regiões invertidas e reposicionadas no mesmo local (ABDOUN, 2012), esta mutação é ilustrada na figura 19.

Figura 19 – Mutação CIM



Fonte: Autor (2018)

3.7 CRITÉRIO DE PARADA

Existem vários critérios de parada para um algoritmo genético, podemos citar:

- i. Máximo de gerações alcançado
- ii. Tempo limite esgotado
- iii. Sem mudança significativa no *fitness* por N gerações.

Em ambos os casos, a busca se encerra e o programa é finalizado, registrando os resultados da forma mais conveniente para o algoritmo (SIVANANDAM; DEEPA, 2007).

4 METODOLOGIA

Para alcançar o fim deste trabalho, é necessário aplicar o AG ao CVRP modificado proposto neste trabalho, para isso primeiro é preciso definir quais são exatamente os operadores e constantes usados na construção do algoritmo, em seguida definir os dados utilizados nos experimentos e, por fim, explicar como foram realizados os experimentos em si.

4.1 CONSTRUÇÃO DO AG

As próximas seções detalharão a construção do AG e os motivos para cada escolha de parâmetro e operador, a função *fitness* também será apresentada. Todas as etapas do algoritmo genético foram implementadas utilizando a linguagem Python em conjunto com a biblioteca NumPy, responsável por otimizar as operações matriciais, os gráficos foram gerados utilizando a biblioteca Matplotlib.

4.1.1 Inicialização

Na inicialização do algoritmo é preciso ter em mãos a codificação do algoritmo e o tamanho da população. Para este estudo foi definido uma população de tamanho 150, parâmetro que foi definido por meio de testes após a finalização do programa, é uma população suficientemente grande para produzir bons resultados sem estender excessivamente o tempo de execução.

Foi escolhida a codificação de permutação, por se tratar de um problema de roteamento, esta codificação reflete a natureza do problema onde cada solução é um reordenamento de uma rota padrão. Para este trabalho foi utilizada uma rota de 60 pontos, onde cada cromossomo é representado por um conjunto em qualquer ordem dos números 1 até 61.

O processo para se gerar estes indivíduos é computacionalmente simples: se gera uma lista contendo os números de 1 até 61, e em seguida se embaralha essa lista, para este propósito foi utilizada a função *shuffle* da biblioteca NumPy.

4.1.2 Seleção

A seleção utilizada foi a de torneio. Existem dois principais motivos para esta escolha:

i. A seleção por torneio proporciona um controle da pressão seletiva na população, permitindo um ajuste por meio do tamanho do torneio.

ii. A seleção por torneio pode acontecer sem se computar o valor do *fitness* para a população inteira a cada geração.

O primeiro item é relevante pois afeta a qualidade das soluções por meio de um controle fino, já o segundo item é importante pois a função *fitness* desenvolvida é computacionalmente custosa.

Para se determinar o tamanho do torneio, executou-se o programa final e o tempo de execução foi analisado junto ao *fitness* médio para uma população reduzida, após isso o tamanho do torneio escolhido foi 6.

4.1.3 Cruzamento

Levando em conta o problema e a codificação por permutação do AG, existem algumas opções disponíveis de operadores de cruzamento, entre elas o cruzamento OX e MPX. O operador escolhido foi o OX e sua escolha se deu principalmente pelo seu custo computacional baixo, visto que o MPX tem uma série de passos a mais e o cruzamento OX, menos custoso, teve um impacto visível no tempo de execução do algoritmo.

A taxa de cruzamento ou *crossover rate* foi escolhida baseada em performance e elitismo: performance porque o cruzamento OX, assim como a função *fitness* é custoso e elitismo pois ao se manter os pais no cruzamento, se mantém indivíduos de alto *fitness* na população sem modificá-los, a taxa escolhida foi de 12%.

4.1.4 Mutação

O operador de mutação escolhido foi o TWORS por se tratar de um operador de mutação simples e não inserir grandes perturbações na população, o papel idealizado para a mutação neste trabalho é o de inserir pequenas modificações, logo outros operadores de mutação seriam mais impactantes que o desejável.

A taxa de mutação escolhida foi de 15%, esta taxa também foi escolhida utilizando-se testes preliminares e o seu critério de escolha foi manter um certo desvio padrão na população.

4.1.5 Critério de Parada

O critério de parada utilizado foi o de número de gerações e foi definido em conjunto com o tamanho da população, tentando maximizar os resultados enquanto o tempo de execução se mantinha abaixo de cinco minutos, sendo 150 o número de gerações escolhido.

4.1.6 Fitness

Para a construção do *fitness* foi utilizada uma simulação de atendimento a demanda, onde se simula um veículo de até 30 passageiros seguindo a rota definida pelo cromossomo, o algoritmo funciona da seguinte forma:

- i. O veículo inicia no primeiro ponto da rota do cromossomo com uma capacidade de até 30 passageiros.
- ii. Para cada iteração o veículo avança para o próximo ponto da rota, adquirindo novos passageiros se sua carga ainda não chegou em 30, para cada novo passageiro a carga sobe em um.
- iii. Para cada iteração o veículo perde cerca de 17% da sua carga.

Esse processo se repete por um número de iterações igual a três vezes o número de pontos na rota, representando três voltas completas na rota. No terceiro passo a perda de carga a cada ponto é baseado na quantidade média de paradas por passageiro (LLANES, 2017).

O valor da *fitness* é equivalente ao número total de passageiros coletados pelo veículo, sendo essa a métrica principal dos resultados.

4.2 ELABORAÇÃO DOS DADOS

Os dados necessários para este algoritmo genético são dois: a quantidade de pontos numa rota e um conjunto de pesos onde cada ponto da rota tem um peso correspondente, o tamanho da rota foi fixado pelo autor em 60 pontos e dois conjuntos de pesos foram gerados por dois métodos diferentes:

i. Para o conjunto de pesos 1 ou distribuição 1, foi usada uma distribuição uniforme com pontos entre 0 e 15, simulando um cenário onde cada ponto é igualmente preenchido com indivíduos.

ii. Para o conjunto de pesos 2 ou distribuição 2, foi utilizada uma distribuição gaussiana com média 5 e desvio padrão 3, o motivo desta escolha é o de simular uma dessimetria na concentração de passageiros nos pontos, sendo este um cenário mais real.

No final deste processo de geração os dois pesos foram salvos em um arquivo de texto para serem utilizados pelo programa. Os conjuntos de peso 1 e 2 estão nos apêndices A e B respectivamente.

4.3 EXPERIMENTOS

Os experimentos foram realizados executando o algoritmo genético dez vezes para cada conjunto de pesos, totalizando vinte execuções onde em cada uma destas execuções se extraiu os valores da média, desvio padrão e o melhor indivíduo em todas as gerações, estas três métricas estão em função do *fitness*, ou seja, passageiros coletados. Ao final deste processo os dados foram divididos em dois grupos: conjuntos de pesos 1 e 2.

Ao ter esses dados em mãos, foi tirado o valor médio das médias, desvios padrão e melhores para todas as execuções, sendo este o dado final para os resultados por cada peso.

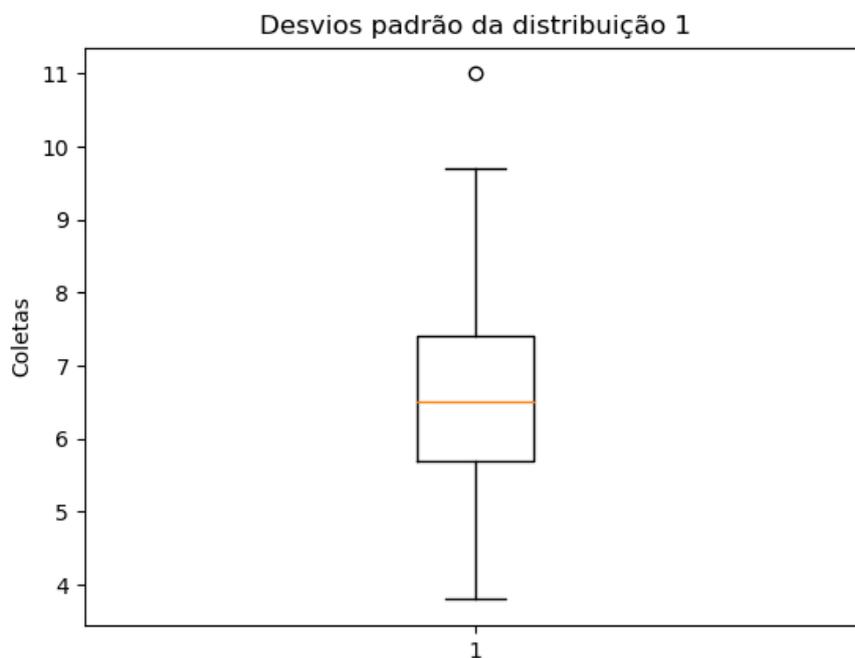
5 RESULTADOS

A análise de resultados se dá em duas partes: a primeira referente ao desempenho do algoritmo genético, com 3 gráficos de diagrama de caixa para cada conjunto de peso, esses três gráficos são construídos utilizando a média, desvio padrão e o maior valor de *fitness* para o conjunto de pesos correspondente, a segunda parte analisa o impacto do algoritmo genético, para isto se cria um gráfico para cada conjunto de peso mostrando os melhores indivíduos de cada conjunto de pesos ordenados de forma crescente com uma indicação da média da rota padrão (a rota que segue do ponto um ao sessenta e um de forma crescente). A unidade para ambas as análises e gráficos é a métrica principal do trabalho, a quantidade de passageiros coletados. Cada execução teve um limite de cinco minutos.

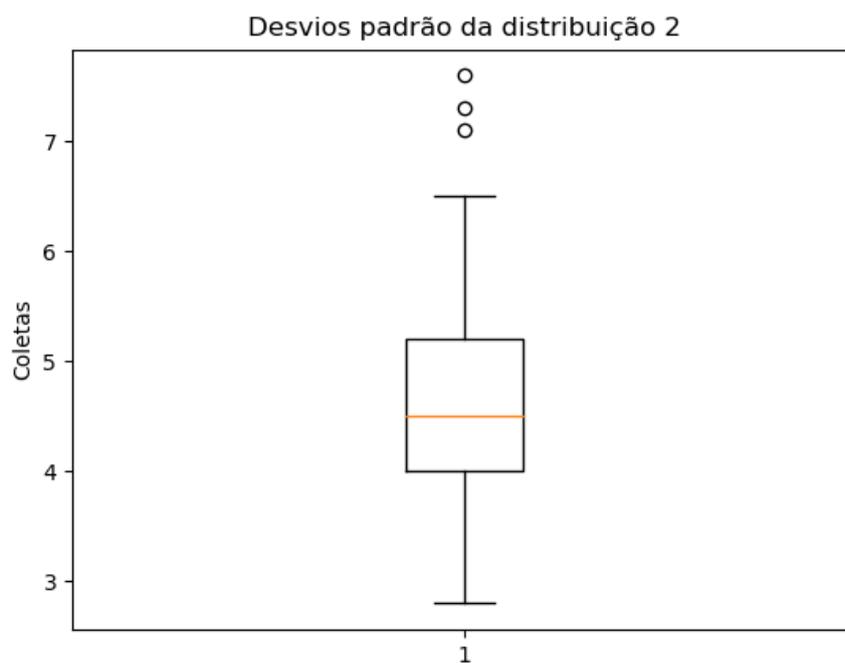
5.1 ANÁLISE DO ALGORITMO GENÉTICO

As análises do algoritmo genético em si foram realizadas a partir de diagramas de caixa, gráficos que mostram com facilidade a variação dos dados, sendo caracterizados por um retângulo atravessado por uma linha vertical, a deformação deste retângulo no eixo Y indica a distribuição dos valores do primeiro ao terceiro quadrante de baixo para cima, a linha indica os valores entre os quadrantes em uma taxa de tolerância baseada na diferença de valor entre o primeiro valor do primeiro quadrante e o último do terceiro, elementos fora desta taxa de tolerância são divergentes da distribuição e são representados por círculos brancos, a unidade do eixo Y é a quantidade de coletas. Sobre o eixo X existe um indicador de valor 1, isso ocorre porque existe só uma distribuição sendo observada por gráfico, este valor não tem significado prático.

A figura 20 ilustra os desvios padrão para o conjunto de pesos 1, onde é possível visualizar que em termos de quantidades de coletas o desvio padrão é baixo, padrão que se repete para o conjunto de pesos 2 na figura 21, pode se verificar alguns elementos destoantes na região superior do gráfico, isso se dá em grande parte na inicialização do algoritmo, onde os indivíduos ainda estão muito dispersos no espaço de busca.

Figura 20 – Desvios padrão da distribuição 1

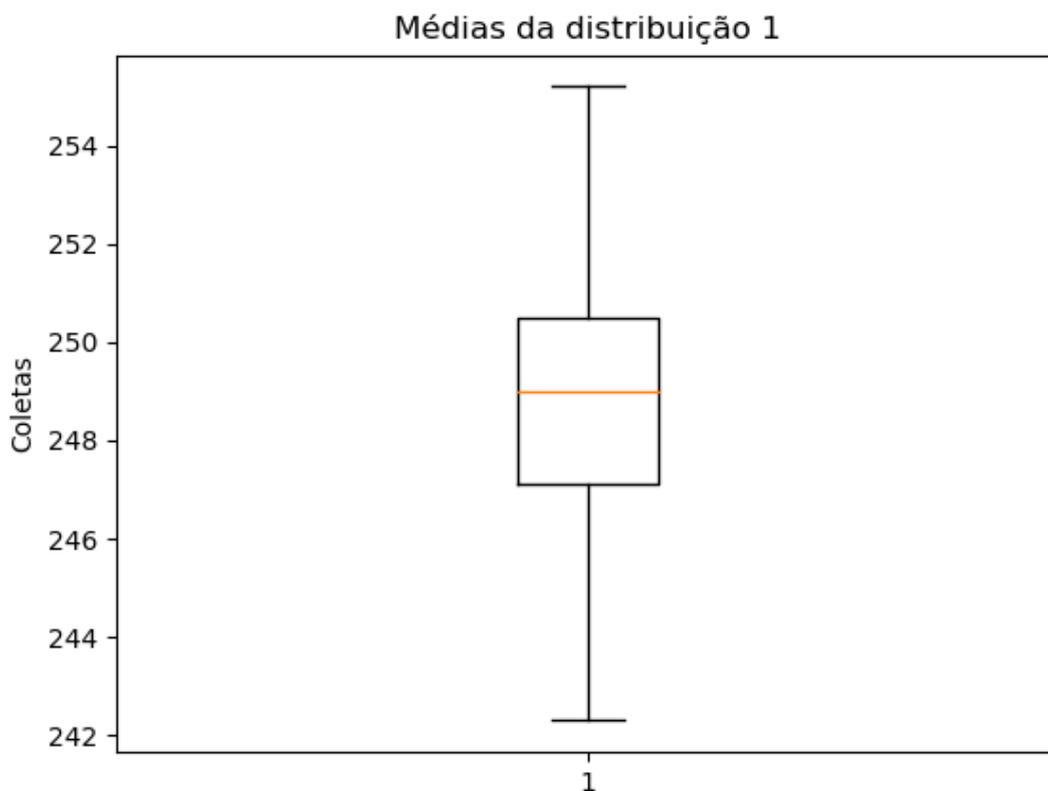
Fonte: Autor (2018)

Figura 21 – Desvios padrão da distribuição 2

Fonte: Autor (2018)

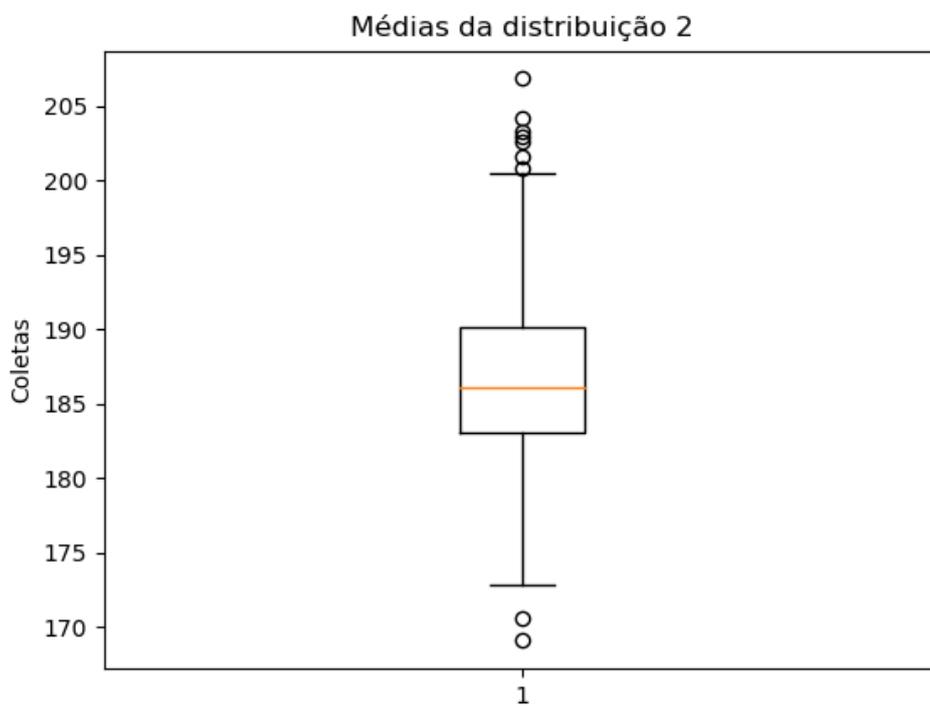
Já quando se trata da média, a figura 22 ilustra os resultados no conjunto de pesos 1, é possível observar que os mesmos orbitaram fortemente em torno de 240 e 260 coletas, sendo que o valor de coletas para uma rota padrão é o de 260. Apesar disso algumas gerações tiveram médias bem acima do valor padrão chegando a valores de até 270 coletas, esse padrão se repete na figura 23, onde o *fitness* de uma rota padrão é de 197 coletas, levemente acima da média das gerações para o conjunto de pesos 2, porém excedida com grande margem por algumas gerações com alto valor de *fitness*.

Se concluir a partir das análises de média que a variabilidade foi menor do que a desejável para uma exploração completa do espaço de busca, já que grande parte dos resultados orbitou um certo *fitness*, mostrando uma certa influência de ótimos locais.

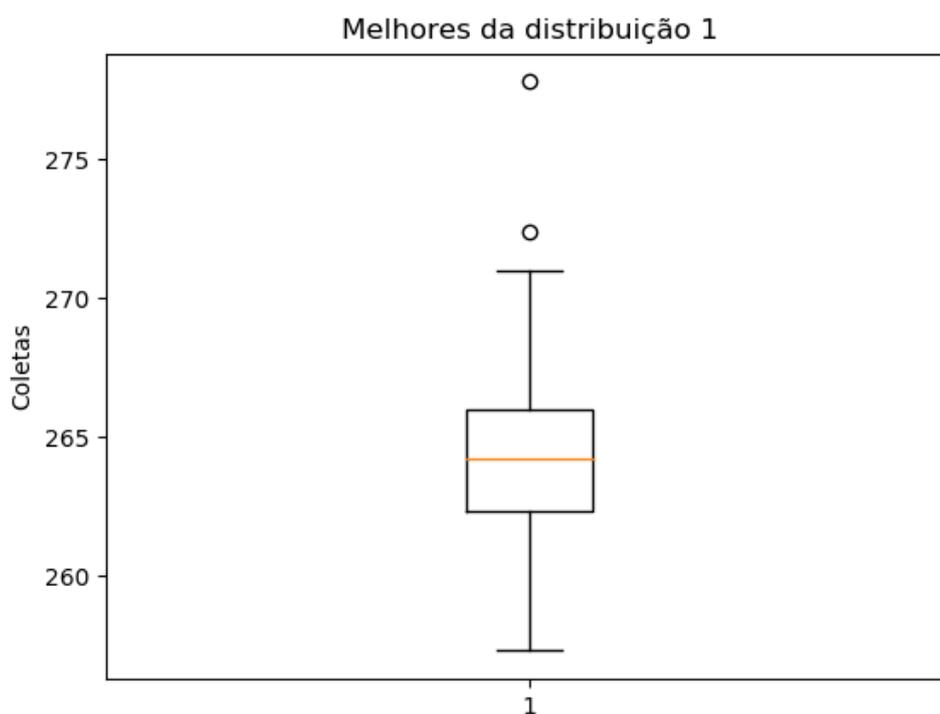
Figura 22 – Médias da distribuição 1

Fonte: Autor (2018)

Por fim, as figuras 24 e 25 mostram a distribuição dos melhores resultados para as distribuições 1 e 2 respectivamente onde é possível observar que a média é bem mais alta que os valores padrão para ambos os pesos, indicando que a busca do algoritmo genético levou a resultados relevantes.

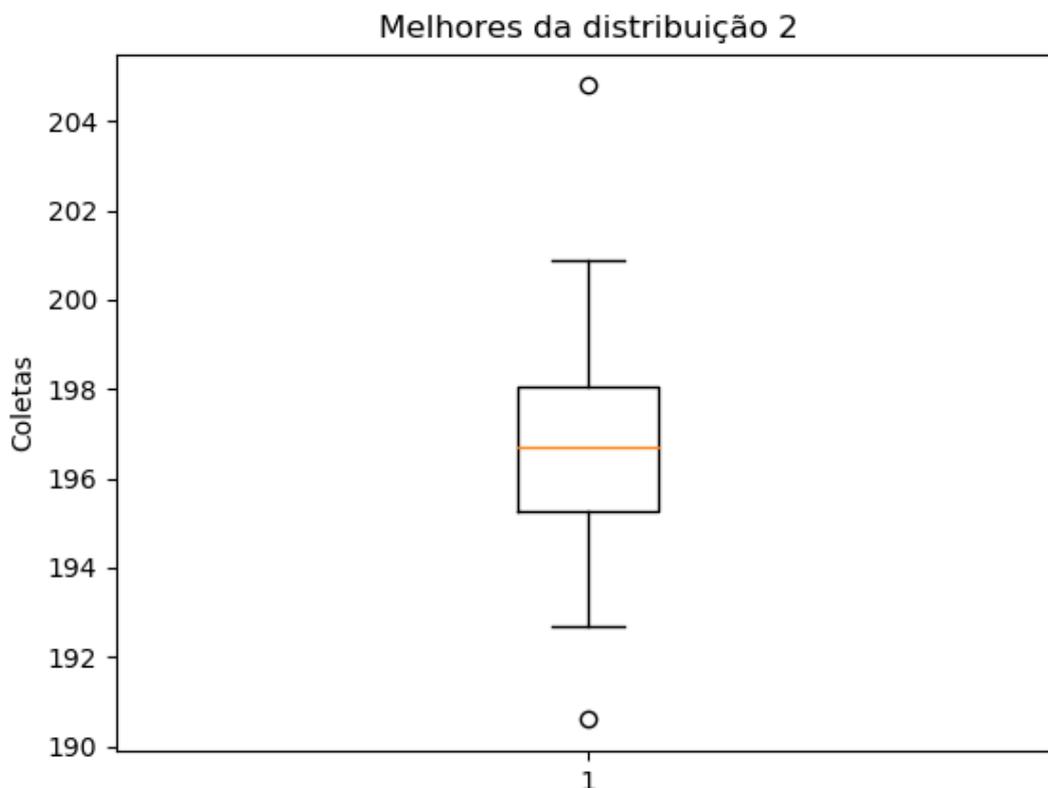
Figura 23 – Médias da distribuição 2

Fonte: Autor (2018)

Figura 24 – Melhores da distribuição 1

Fonte: Autor (2018)

Figura 25 – Melhores da distribuição 2

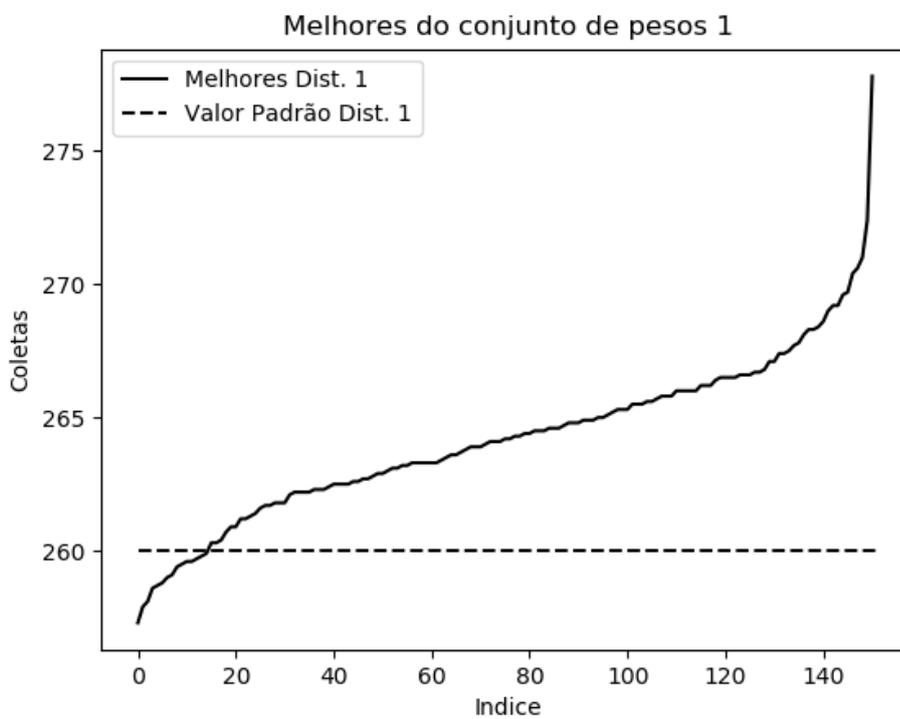


Fonte: Autor (2018)

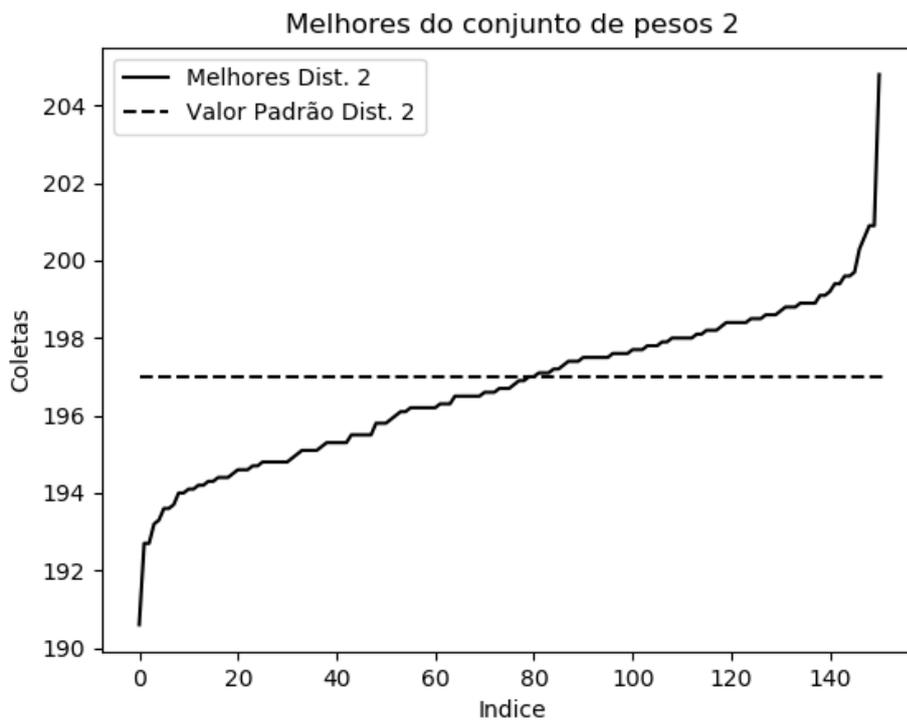
5.2 ANÁLISE DOS RESULTADOS

Para as figuras 26 e 27, temos uma comparação dos melhores indivíduos de cada geração em ordem crescente em termos de número de coletas, para facilitar a visualização da relação entre esses valores e a rota padrão, foi inserida também a quantidade de coletas para a rota padrão por meio de uma linha tracejada horizontal, as figuras 26 e 27 representam respectivamente esse processo feito para o conjunto de pesos um e dois, em ambos os casos é possível observar uma pequena quantidade de gerações onde o melhor tem *fitness* menor que a rota padrão, porém grande parte das gerações tem um desempenho superior.

Algo a se considerar também é como para o conjunto de pesos 1, existe uma grande quantidade de boas soluções e uma curva de crescimento rápido com grande parte dos indivíduos em torno da média porém com uma tendência à direita, já no conjunto de pesos 2 a média de *fitness* se desloca consideravelmente à esquerda de forma que a maioria dos indivíduos estão com um *fitness* próximo a média.

Figura 26 – Melhores do conjunto de pesos 1

Fonte: Autor (2018)

Figura 27 – Melhores do conjunto de pesos 2

Fonte: Autor (2018)

6 CONCLUSÃO

Os resultados apontam que a utilização do algoritmo genético foi relevante para o estudo de rotas com pesos associados gerando resultados bons com consistência, no entanto essa pesquisa ainda pode ser aprimorada de algumas formas. A utilização do algoritmo genético foi relevante para o estudo de rotas com pesos associados, gerando boas rotas com consistência. No entanto, ainda existem pontos onde esta pesquisa pode ser aprimorada, entre eles podemos citar a utilização e comparação de outros algoritmos para resolver o problema, uma elaboração do modelo computacional mais consistente com o problema real e o aprimoramento das simulações utilizadas nos resultados e na função *fitness*.

O algoritmo utilizado neste trabalho foi apenas o algoritmo genético, comparações com outros algoritmos, tanto exatos quanto metaheurísticas, pode trazer dados mais relevantes ao estudo.

A modelagem computacional do problema, um CVRP modificado, pode ser aprimorada para refletir com mais consistência o problema real do transporte, assim como a simulação utilizada nos resultados e no *fitness* de cada indivíduo pode se beneficiar do mesmo processo. Uma melhoria em qualquer aspecto poderia resultar em dados mais relevantes do ponto de vista científico.

6.1 DIFICULDADES ENCONTRADAS

O algoritmo foi uma grande fonte de dificuldades, primeiramente no desenvolvimento e em seguida na otimização dos parâmetros, o operador *fitness* precisa ser executado milhares de vezes durante a execução e para isso ele foi otimizado diversas vezes. A análise dos resultados também foi uma fonte de dificuldades pela geração dos gráficos e interpretação dos mesmos.

6.2 TRABALHOS FUTUROS

Apesar de apresentar resultados satisfatórios, este trabalho pode ser aprimorado em alguns aspectos, dentre eles:

- i. A utilização e comparação de outros algoritmos para a solução do problema pode permitir uma melhor análise de resultados, o algoritmo genético se provou útil para o problema em questão, porém uma análise de outros métodos é desejável.

ii. O aprimoramento do modelo teórico que representa o problema em questão, trazendo mais fidelidade à modelagem em comparação com o cenário real.

iii. O aprimoramento da simulação pode trazer um desempenho melhor para a fase de fitness, trazendo resultados mais rapidamente e proporcionando uma exploração maior do espaço de busca no caso do algoritmo genético, além de proporcionar um cenário mais próximo do cenário real, como no caso da modelagem.

7 REFERÊNCIAS

ABDOUN, Otman; ABOUCHABAKA, Jaafar; TAJANI, Chakir. Analyzing the performance of mutation operators to solve the travelling salesman problem. **arXiv preprint arXiv:1203.3099**, 2012.

AMERICAN PUBLIC TRANSPORTATION ASSOCIATION et al. Economic Impact of Public Transportation Investment: 2014 Update. **Washington, DC: Economic Development Research Group, Inc. (EDRG)**, 2014.

BANISTER, David. **Transport and urban development**. Routledge, 2003.

GOMIDE, Alexandre de Ávila. **Mobilidade urbana, iniquidade e políticas sociais**. 2006.

HALL, Jonathan D. et al. **Is Uber a substitute or complement for public transit?**. 2017.

HOLLAND, John Henry. **Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence**. MIT press, 1992.

LAPORTE, Gilbert; NOBERT, Yves. Exact algorithms for the vehicle routing problem. In: **North-Holland Mathematics Studies**. North-Holland, 1987. p. 147-184.

_____. The traveling salesman problem: An overview of exact and approximate algorithms. **European Journal of Operational Research**, v. 59, n. 2, p. 231-247, 1992.

LLANES, Kathrin Rodríguez et al. An Approach to Evaluate the Impact on Travel Time of Bus Network Changes. In: **Proceedings of the International Conference of Enterprise Information Systems (ICEIS)**. 2017.

LI, Z.; HONG, Y.; ZHANG, Z. **Do on-demand Ride-Sharing Services Affect Traffic Congestion? Evidence from Uber Entry**. Working paper, available at SSRN: <https://ssrn.com/abstract=2838043>, 2016.

LIN, Jyh-Horng; CHOU, Ti-Chieh. A geo-aware and VRP-based public bicycle redistribution system. **International Journal of Vehicular Technology**, v. 2012, 2012.

MCCALL, John. Genetic algorithms for modelling and optimisation. **Journal of Computational and Applied Mathematics**, v. 184, n. 1, p. 205-222, 2005.

MEMARI, Ashkan; AHMAD, Robiah; RAHIM, Abd Rahman Abdul. Metaheuristic Algorithms: Guidelines for Implementation. **Journal of Soft Computing and Decision Support Systems**, v. 4, n. 6, p. 1-6, 2017.

MITCHELL, Melanie. **An introduction to genetic algorithms**. MIT press, 1998.

MOSCATO, Pablo et al. On genetic crossover operators for relative order preservation. **C3P Report**, v. 778, 1989.

ROZESTRATEN, R. J. A. **Psicologia do Trânsito – Conceitos e Processos Básicos**. 4. ed. São Paulo: Makron Books, 1996.

SIVANANDAM, S. N.; DEEPA, S. N. **Introduction to genetic algorithms**. Springer Science & Business Media, 2007.

SNYMAN, Jan. **Practical mathematical optimization: an introduction to basic optimization theory and classical and new gradient-based algorithms**. Springer Science & Business Media, 2005.

VIANNA, Guilherme Szczerbacki Besserman; YOUNG, Carlos Eduardo Frickmann. EM BUSCA DO TEMPO PERDIDO: UMA ESTIMATIVA DO PRODUTO PERDIDO EM TRÂNSITO NO BRASIL. **Revista de Economia Contemporânea**, v. 19, n. 3, p. 403-416, 2015.

APÊNDICE A – Conjunto de pesos 1

Abaixo estão os valores de peso do conjunto 1 na ordem crescente de índice na rota, gerados utilizando a distribuição uniforme.

5, 3, 3, 10, 13, 6, 9, 4, 0, 2, 12, 4, 13, 2, 8, 13, 12, 8, 8, 2, 1, 4, 4, 9, 7, 5, 13, 10, 4, 2, 2, 12, 8, 0, 7, 8, 5, 2, 1, 8, 5, 13, 12, 11, 0, 13, 0, 13, 1, 6, 8, 8, 2, 13, 7, 15, 6, 12, 3, 12.

APÊNDICE B – Conjunto de pesos 2

Abaixo estão os valores de peso do conjunto 1 na ordem crescente de índice na rota, gerados utilizando a distribuição gaussiana.

5, 6, 10, 1, 3, 2, 5, 13, 4, 4, 7, 1, 6, 4, 3, 0, 5, 2, 5, 5, 9, 1, 10, 0, 5, 5, 4, 6, 1, 5, 5, 1, 5,
1, 6, 6, 4, 5, 10, 2, 14, 1, 4, 6, 3, 4, 5, 7, 9, 9, 9, 3, 6, 3, 8, 9, 9, 7, 3.