

UNIVERSIDADE FEDERAL DE ALAGOAS - UFAL
CAMPUS A. C. SIMÕES
CIÊNCIA DA COMPUTAÇÃO

JOSÉ ERALDO DOS SANTOS NETO

**DESENVOLVIMENTO DE UM MODELO DE ALGORITMO GENÉTICO PARA
OTIMIZAÇÃO DO CUSTO NA FASE DE TRANSPORTE LOGÍSTICO DO
PEQUENO EMPREENDEDOR**

MACEIÓ
2023

José Eraldo dos Santos Neto

Desenvolvimento de um modelo de Algoritmo Genético para otimização do custo na fase de
Transporte Logístico do Pequeno Empreendedor

Monografia apresentada como requisito
parcial para obtenção do grau de
Bacharel em Ciência da Computação da
Universidade Federal de Alagoas –
UFAL, Campus A. C. Simões.

Orientadora: Profa. Dra. Roberta
Vilhena Vieira Lopes.

Maceió

2023

Catálogo na fonte
Universidade Federal de Alagoas
Biblioteca Central
Divisão de Tratamento Técnico
Bibliotecária: Taciana Sousa dos Santos – CRB-4 – 2062

S237d Santos Neto, José Eraldo dos.

Desenvolvimento de um modelo de algoritmo genético para otimização do custo na fase de transporte logístico do pequeno empreendedor / José Eraldo dos Santos Neto. – 2023.

52 f. : il. color.

Orientadora: Roberta Vilhena Vieira Lopes.

Monografia (Trabalho de Conclusão de Curso em Ciência da Computação) – Universidade Federal de Alagoas. Instituto de Computação. Maceió, 2023.

Bibliografia: f. 48-50.

Apêndice: f. 51-52.

1. Algoritmos genéticos. 2. Otimização. 3. Modelo computacional. 4. Logística. I. Título.

CDU: 004.021

AGRADECIMENTOS

Agradeço a Deus por toda a sua graça e por todas as coisas que Ele tem proporcionado em meu caminho pessoal, profissional e da vida como um todo.

Aos meus pais, que me deram todo o conforto, educação de qualidade e apoio em toda a minha vida. Também, aos meus avós maternos, Estela e Eraldo, por todo amor e carinho. Gostaria que todos vocês fossem eternos. À minha avó paterna, Irene, que não está mais nesse plano, por todos os puxões de orelha e suas histórias. Sinto a sua falta.

A minha namorada, Yana Gabriella, por todo amor e apoio durante todos os momentos.

A professora Roberta, que dedicou seu tempo para me orientar no final da graduação e me ensinar conceitos importantes, os quais tive a honra de aprender nas disciplinas de Estrutura de Dados e Computação Evolucionária.

A banca de professores, pela presença e tempo para avaliar o meu trabalho.

Aos meus colegas de curso e amigos da vida, Gláucio Honório e William Kleber, que me acompanharam no início da graduação e a tornaram mais amigável. “Os caras da programação” é uma memória que eu guardo com muito afeto.

Aos funcionários do Instituto de Computação: Ana Ferreira, Marcelo de Gusmão, Simone e Laelson; que me ajudaram com as questões burocráticas durante toda a graduação.

A todos os professores do Instituto de Computação que dedicam suas vidas para educar e formar profissionais com qualidade, em especial: Rafael de Amorim, Fábio Paraguaçu e Balduino Fonseca. Aos professores Anamelea de Campos Pinto (*in memoriam*) e Luís Paulo L. Mercado pela introdução à vida acadêmica no Centro de Educação, como bolsista PIBIC.

A todos aqueles que não mencionei, mas que estão guardados na memória com afeto.

Eraldo Neto

“They dared to go where no one would try, they chose to fly where eagles dare.”

Where Eagles Dare, Iron Maiden.

RESUMO

Os Algoritmos Genéticos são algoritmos meta-heurísticos de busca, que simulam o processo evolutivo descrito na Teoria da Evolução de Charles Darwin. De acordo com a teoria, os indivíduos com melhores adaptações ao ambiente são aqueles que sobrevivem. Assim, essa ideia foi aplicada no cenário da computação por John H. Holland, que em 1975 criou um modelo de computação baseado na evolução biológica para solucionar problemas de otimização. A simplicidade do seu algoritmo permitiu buscar soluções aproximadas para problemas NP (Não-Polinomiais), tentando encontrar uma solução ótima, mas sem garantias de que a solução encontrada seja a melhor. A busca da solução do problema é construída por meio de uma série de funções que ele definiu como operadores genéticos: criação de população, seleção dos mais adaptados, combinação, mutação e substituição. A cada processo, é criada uma população descendente que pode ser a solução potencial, mas que passará pela etapa de avaliação da função objetivo até que uma solução satisfatória seja encontrada. Devido a sua flexibilidade e capacidade de solucionar problemas, os algoritmos genéticos podem ser utilizados em diversas áreas do conhecimento, incluindo a Teoria dos Jogos, Inteligência Artificial, finanças, Ciência de Dados e Engenharias, com a otimização de processos, modelos e aprendizado. No Brasil, os microempreendedores e pequenas empresas enfrentam desafios significativos no setor logístico, reflexo da falta de investimento por parte do governo, baixo capital para planejamentos estratégicos e infraestrutura deficiente. Isso resulta no aumento dos custos de operações, sobretudo na fase do transporte de mercadorias. Dessa forma, este trabalho busca aplicar o conceito de algoritmos genéticos para o desenvolvimento de um modelo computacional evolutivo que auxilie na redução dos custos operacionais logísticos de pequenos empreendedores, na fase de transporte de mercadorias.

Palavras-chave: Algoritmos Genéticos. Logística. Otimização. Pequeno Empreendedor.

ABSTRACT

Genetic algorithms are metaheuristic search algorithms that simulate the evolutionary process described in Charles Darwin's theory of evolution. According to the theory, individuals with better adaptations to the environment are the ones that survive. Thus, this idea was applied to the computing scenario by John H. Holland, who in 1975 created a computational model based on biological evolution to solve optimization problems. The simplicity of his algorithm allowed for approximate solutions to NP (non-polynomial) problems to be found, attempting to find an optimal solution, but without guarantees that the solution found will be the best. The search for the problem solution is constructed through a series of functions that he defined as genetic operators: population creation, selection of the fittest, combination, mutation, and substitution. At each process, a descendant population is created that could be the potential solution, but it will go through the objective function evaluation stage until a satisfactory solution is found. Due to its flexibility and problem-solving ability, genetic algorithms can be used in various fields of knowledge, including Game Theory, Artificial Intelligence, finance, Data Science, and Engineering, optimizing processes, models, and learning. In Brazil, microentrepreneurs and small businesses face significant challenges in the logistics sector, reflecting the lack of government investment, low capital for strategic planning, and deficient infrastructure. This results in increased operational costs, particularly in the transportation phase of goods. Thus, this work aims to apply the concept of genetic algorithms to the development of an evolutionary computational model that assists in reducing the logistics operational costs of small entrepreneurs in the transportation phase of goods.

Keywords: Genetic Algorithms. Logistics. Optimization. Small Entrepreneur.

LISTA DE ABREVIATURAS

AG	Algoritmo Genético
GEM	Global Entrepreneurship Monitor
SUS	Stochastic universal sampling
TAE	Taxa de Atividade Empreendedora
VRP	Vehicle Routing Problem
TSP	Traveler Salesman Problem
PSO	Particle Swarm Optimization

Símbolos:

n - representa uma quantidade genérica de indivíduos.

p - população do algoritmo genético de Holland

h - número de cromossomos selecionados para gerar descendentes no algoritmo genético de Holland

$\hat{p}$ - população de cromossomos selecionados pela ação dos operadores genéticos de cruzamento e mutação no algoritmo de Holland

$k^{\rightarrow}$ - vetor de bits

$\alpha^{\rightarrow}$ - cromossomo genérico utilizado para representar um problema qualquer

$p^{\rightarrow}$ - vetor n -dimensional que representa uma população de n $k^{\rightarrow}$

P_i - probabilidade de seleção, calculada pela relação de aptidão do cromossomo dividido pela soma de todas as aptidões.

$f(x_i)$ - função de adaptação pré-definida que calcula o valor de adaptação de um cromossomo

$\sum_{k=1}^n f(k)$ - soma de todos os valores de aptidão de uma população de n indivíduos

$rank$ - função que mapeia os indivíduos de acordo com a sua posição na classificação

p_c - probabilidade de ocorrer a operação de cruzamento em uma população de cromossomos

p_m - probabilidade de ocorrer a operação de mutação em uma população de cromossomos

t - tamanho de um cromossomo $k^{\rightarrow}$

g - gene de um cromossomo $\alpha^{\rightarrow}$

b - bit representado em um gene

φ - conjunto formado pelos valores dos genes de $\alpha^{\rightarrow}$

LISTA DE FIGURAS

Figura 1 - Pseudocódigo de um algoritmo genético clássico.....	8
Figura 2 - Representação de um array de bits.....	19
Figura 3 - População de cromossomos.....	20
Figura 4 - Cálculo de adaptação em uma população.....	21
Figura 5 - Pseudocódigo da seleção por roleta.....	22
Figura 6 - Método de seleção por roleta.....	22
Figura 7 - Método de seleção por classificação.....	24
Figura 8 - Operação de cruzamento.....	27
Figura 9 - Duplo ponto de corte.....	28
Figura 10 - Mutação por inversão de bit.....	29
Figura 11 - Mutação por inserção.....	29
Figura 12 - Mutação uniforme.....	30
Figura 13 - Conjunto de gráficos da simulação 5.....	41
Figura 14 - Conjunto de gráficos da simulação 10.....	41
Figura 15 - Gráficos da evolução da população.....	42
Figura 16 - Gráficos da evolução da população	43
Figura 17 - Gráficos da evolução da população	44
Figura 18 - Gráficos da evolução da população.....	45
Figura 19 - Gráficos das rotas inicial e final.....	46
Figura 20 - Executar o modelo.....	52
Figura 21 - Alterar parâmetros do modelo.....	52
Figura 22 - Executar o modelo.....	53

LISTA DE TABELAS

Tabela 1 - Probabilidade de seleção.....	22
Tabela 2 - Simulação do modelo evolucionário.....	39
Tabela 3 - Continuação das simulações do modelo evolucionário.....	40
Tabela 4 - Continuação das simulações do modelo evolucionário.....	41
Tabela 5 - Continuação das simulações do modelo evolucionário.....	42
Tabela 6 - Continuação das simulações do modelo evolucionário.....	43
Tabela 7 - Últimas simulações do modelo evolucionário.....	44

SUMÁRIO

1. INTRODUÇÃO.....	12
1.1 JUSTIFICATIVA.....	13
1.2 OBJETIVO GERAL.....	13
1.3 OBJETIVOS ESPECÍFICOS.....	14
1.4 METODOLOGIA.....	14
1.5 ORGANIZAÇÃO DO DOCUMENTO.....	15
2. REFERENCIAL TEÓRICO.....	16
2.1 REPRESENTAÇÃO DO CROMOSSOMO.....	18
2.2 REPRESENTAÇÃO DA POPULAÇÃO.....	19
2.3 FUNÇÃO DE ADAPTAÇÃO.....	20
2.4 OPERADOR GENÉTICO DE SELEÇÃO.....	21
2.4.1 MÉTODO DE SELEÇÃO POR ROLETA.....	21
2.4.2 MÉTODO DE SELEÇÃO POR CLASSIFICAÇÃO.....	23
2.4.3 MÉTODO DE SELEÇÃO POR TORNEIO.....	24
2.4.4 MÉTODO DE SELEÇÃO ELITISTA.....	25
2.5 OPERADOR GENÉTICO DE CRUZAMENTO.....	26
2.6 OPERADOR GENÉTICO DE MUTAÇÃO.....	27
2.6.1 MUTAÇÃO POR INVERSÃO DE BIT.....	27
2.6.2 MUTAÇÃO POR INSERÇÃO.....	28
2.6.3 MUTAÇÃO POR TROCA.....	29
2.6.4 MUTAÇÃO UNIFORME.....	29
2.7 OPERADOR GENÉTICO DE SUBSTITUIÇÃO.....	30
2.8 PROBABILIDADE DE CRUZAMENTO E MUTAÇÃO.....	30
3. TRABALHOS RELACIONADOS.....	31
3.1 O PROBLEMA DO CAIXEIRO VIAJANTE.....	31
3.2 PROBLEMA DE ROTEAMENTO DE VEÍCULOS.....	32
3.3 O RECOZIMENTO SIMULADO.....	34
3.2 O ENXAME DE PARTÍCULAS.....	35
3.2 CONSIDERAÇÕES.....	35
4. O MODELO.....	36
4.1 DEFINIÇÕES TÉCNICAS DO MODELO.....	36
4.2 ESPECIFICAÇÕES DO MODELO GENÉTICO.....	36
4.2.1 DEFINIÇÃO DO PROBLEMA.....	37
4.2.2 REPRESENTAÇÃO DE SOLUÇÕES.....	37
4.2.3 REPRESENTAÇÃO DO CUSTO.....	37
4.2.4 FUNÇÃO OBJETIVO.....	38
4.2.5 OPERADORES GENÉTICOS.....	38
4.2.6 CRITÉRIO DE PARADA.....	39
4.2.7 VALIDAÇÃO DO MODELO.....	39

4.3 RESULTADOS E DISCUSSÕES.....	39
4.3.1 PRIMEIRA ETAPA DE SIMULAÇÃO.....	40
4.3.2 SEGUNDA ETAPA DE SIMULAÇÃO.....	41
4.3.3 TERCEIRA ETAPA DE SIMULAÇÃO.....	42
4.3.4 QUARTA ETAPA DE SIMULAÇÃO.....	43
4.3.5 QUINTA ETAPA DE SIMULAÇÃO.....	44
4.3.6 SEXTA ETAPA DE SIMULAÇÃO.....	45
5. CONCLUSÃO.....	47
REFERÊNCIAS.....	49
APÊNDICE A.....	52

1. INTRODUÇÃO

O crescente progresso da tecnologia tem levado ao aprimoramento constante dos métodos de resolução de problemas e ao desenvolvimento de diversos modelos computacionais para fornecer soluções eficazes para otimização de sistemas. Entre as abordagens mais utilizadas, destacam-se os algoritmos genéticos, que possuem aplicações nos campos da biologia, engenharia e economia, por exemplo.

Esse método, fundamentado nos princípios da seleção natural e da genética populacional, permite a resolução de problemas complexos, por meio de processos evolutivos simulados em um ambiente computacional. Sua capacidade única de buscar soluções completas e globais para um determinado problema faz dos algoritmos genéticos uma poderosa ferramenta de solução de problemas que reúne características tão importantes quanto robustez, versatilidade e simplicidade (BOURMISTROVA, A. KHANTSIS, S. 2010).

Durante o desenvolvimento do conhecimento, os cientistas se debruçaram em observações de fenômenos naturais, com o objetivo de descrever o comportamento dos sistemas por meio de relações matemáticas. Neste contexto, Isaac Newton publicou em 1687 o seu livro "Philosophiæ Naturalis Principia Mathematica" (NEWTON, 1687), no qual criou as teorias da mecânica clássica, como a Lei da Gravitação Universal e a Lei dos Movimentos, que são objetos de estudo até os dias atuais. Da mesma forma, Albert Einstein desenvolveu a Teoria da Relatividade, que foi publicada em 1905 (EINSTEIN, 1905), a partir dos questionamentos da natureza do tempo e do espaço. Assim, nota-se que a natureza tem um papel significativo no que tange a construção da ciência e do saber das coisas.

A computação é uma área do conhecimento que também tem seus alicerces baseados em objetos naturais, dos quais foram desenvolvidos os estudos no campo da Inteligência Artificial, como as redes neurais e sistemas especialistas, por exemplo. A Inteligência Computacional é uma área da ciência que busca, através de técnicas inspiradas na Natureza, o desenvolvimento de sistemas inteligentes que imitam aspectos do comportamento humano, tais como: aprendizado, percepção, raciocínio, evolução e adaptação. [...] As técnicas da Inteligência Computacional têm sido empregadas com sucesso no desenvolvimento de sistemas inteligentes de previsão, suporte à decisão, controle, otimização, modelagem, classificação e reconhecimento de padrões em geral, aplicados em diversos setores: Energia, Industrial, Econômico, Financeiro, Comercial e Outros, Síntese de Circuitos, Meio Ambiente, entre outros (PACHECO. M. A. C, 1999).

O contexto histórico do surgimento dos algoritmos genéticos remete à década de 1970, na Universidade de Michigan, quando o estudante de doutorado John H. Holland, que estava trabalhando em fenômenos gerados por sistemas adaptativos complexos, tentava desenvolver um método computacional. No decorrer do seu trabalho, Holland percebeu que existia uma nítida semelhança entre os fenômenos que estudava e o processo de evolução das espécies, pois assim como a interação entre os agentes adaptativos determinava o resultado dos fenômenos investigados por ele, a interação entre os fatores ambientais determinava a próxima população de uma dada região. Com base nesta constatação, Holland propôs um método computacional para simular o processo de evolução das espécies, denominado de algoritmo genético (VIEIRA, 2003).

O termo empreendedorismo refere-se a capacidade e disposição que uma pessoa tem de identificar uma oportunidade de negócio, que pode ou não ser inovadora, e depositar recursos para que sua ideia seja desenvolvida, assumindo os riscos para que se torne uma empresa bem sucedida. [...] a pesquisa GEM realizada em 2007 revelou que a Taxa de Atividade Empreendedora do Brasil é 39% maior que a média mundial, posicionando o país em nono lugar no ranking dos 42 países que participaram da pesquisa (SOUZA et al., 2010).

Os dados do relatório GEM (2021) mostram que o Brasil ainda é considerado um dos países com maior TAE no mundo, com cerca de 23,5% de novos empreendedores, sendo que 54,7% são mulheres. O grande número de empreendedores brasileiros reflete a necessidade de abrir um negócio motivado pela necessidade, classificando-os como nascentes ou pequenos empreendedores.

Para o pequeno empreendedor, a eficiência logística representa um fator crítico para o sucesso de sua empresa, uma vez que seus recursos financeiros e humanos são limitados. Nesse sentido, um processo logístico eficiente e com baixo custo pode representar uma vantagem competitiva significativa em relação a outras empresas, permitindo um melhor posicionamento no mercado e resultados financeiros mais positivos.

A contextualização acima demonstra claramente a necessidade de aprimorar a eficiência logística no Brasil. Desse modo, o presente trabalho propõe o desenvolvimento de um modelo de otimização baseado em Algoritmo Genético para fornecer rotas logísticas eficientes na fase de transporte de insumos para os pequenos empreendedores. Este modelo, portanto, tem como propósito auxiliar as pequenas empresas a reduzirem seus custos logísticos e aumentarem a sua eficiência na gestão de sua cadeia de suprimentos.

1.1 JUSTIFICATIVA

A logística compreende o conjunto das atividades relacionadas a planejamento, controle e execução de atividades como transporte e distribuição de bens e insumos, manutenção e armazenagem de estoques e processamento de pedidos (FERREIRA JR.; TEIXEIRA, 2007). Dentre os processos da logística, o transporte é a atividade mais importante e representa a maior parte dos custos. Atualmente existem numerosos problemas relacionados à infraestrutura brasileira e logo, desafios a serem solucionados. (MORAIS, G.; CARVALHO, 2022). É importante destacar que as dificuldades enfrentadas na logística não se limitam apenas aos empresários de grande porte, mas também são sentidas pelos pequenos negócios. Segundo o SEBRAE (2023), os pequenos negócios são divididos de acordo com o faturamento anual e engloba as seguintes categorias: Microempreendedor Individual, com o faturamento anual de até R\$ 81 mil; Microempresa, com o faturamento anual de até R\$ 360 mil; Empresa de Pequeno Porte, com o faturamento anual entre R\$ 360 mil e R\$ 4,8 milhões; e o Pequeno Produtor Rural, com propriedade com até 4 módulos fiscais ou faturamento anual de até R\$ 4,8 milhões. Neste trabalho, considerou-se os pequenos empreendedores como as categorias de microempreendedor e microempresa devido a quantidade de recursos limitados que estão envolvidos nos seus negócios.

A baixa quantidade de recursos para o investimento em tecnologia, altíssimos custos com combustíveis e dificuldade no planejamento eficiente para a fretagem de mercadorias fazem com que esses profissionais tenham uma concorrência desleal com o mercado. A minimização dos custos tende a elevar a lucratividade das empresas aumentando a remuneração aos fatores de produção, o que estimula investimentos e contribuem para a geração de empregos. (MARTINS; CAIXETA FILHO, 2009). Então, desenvolver um modelo de solução que possibilite uma maior eficiência no transporte e a redução dos custos enfrentados por esses empreendedores - proposta deste trabalho - pode ser o caminho para o crescimento e o sucesso de pequenas empresas, além de contribuir para o desenvolvimento econômico do país e aumentar a competitividade no mercado.

1.2 OBJETIVO GERAL

Esse trabalho tem como objetivo principal o desenvolvimento de um modelo de algoritmo genético capaz de contextualizar o cenário logístico na fase de transporte de insumos

de pequenos empreendedores, com ênfase na redução dos custos operacionais que estão relacionados.

1.3 OBJETIVOS ESPECÍFICOS

- Reduzir o custo operacional logístico da fase de transporte;
- Fornecer rotas eficientes e seguras para a distribuição de insumos em um território;
- Modelo de algoritmo para o desenvolvimento de uma API gratuita para gerar rotas e diminuir custos, com foco no pequeno empreendedor;
- Maximizar a disponibilidade do setor logístico da empresa, de modo que a fase de distribuição ocorra em um menor espaço de tempo.

1.4 METODOLOGIA

Para o desenvolvimento do texto, pretende-se encontrar o estado da arte sobre Algoritmos Genéticos, estudando os principais autores do tema para compor o referencial teórico e a revisão de literatura, tais como John H. Holland, David E. Goldberg e obras que abordaram o conteúdo; entender os desafios que o pequeno empreendedor possui na fase de distribuição de insumos e estudar a fase logística de transporte para a contextualização e domínio de aplicação do tema.

A construção do modelo se dará por meio da implementação de um AG capaz de fornecer a rota que possui menor custo na fase de transporte logístico do PE, de modo a otimizar os custos totais desta fase de operação de acordo com as especificações do contexto abordado (capítulo 4).

O modelo deverá possuir a seguinte estrutura:

1. **Definição do problema:** É importante definir claramente o problema que se deseja resolver. Isso inclui identificar as variáveis relevantes, as restrições e os objetivos do problema.
2. **Representação de soluções:** Escolher uma representação adequada para as soluções do problema e do custo.

3. **Função objetivo:** É preciso definir uma função objetivo clara que possa ser otimizada pelo algoritmo. Isso envolve maximizar ou minimizar uma determinada métrica que representa o objetivo do problema.
4. **Operadores genéticos:** É necessário definir os operadores genéticos, que são os mecanismos de reprodução, mutação e seleção que permitem que o algoritmo evolua soluções melhores ao longo do tempo.
5. **Crítérios de parada:** É importante definir critérios de parada para o algoritmo, que determinam quando o algoritmo deve parar de iterar. Isso evita que o algoritmo continue a rodar indefinidamente e permite que sejam encontradas soluções satisfatórias dentro de um tempo razoável.
6. **Validação do modelo:** Por fim, é necessário validar o modelo de algoritmo, por meio de testes em dados reais ou por meio de simulações. Isso ajuda a verificar se o modelo está funcionando conforme o esperado e se está produzindo resultados úteis para o problema abordado.

1.5 ORGANIZAÇÃO DO DOCUMENTO

Os aspectos básicos sobre Algoritmos Genéticos são tratados no capítulo 2. O capítulo 3 explora alguns problemas clássicos da computação relacionados a este trabalho, que são analisados com o intuito de mostrar semelhanças e diferenças dentro do contexto do modelo proposto. O capítulo 4 refere-se ao desenvolvimento do modelo e seus resultados. E, finalmente, o capítulo 5 descreve as considerações finais sobre o trabalho.

2. REFERENCIAL TEÓRICO

Os Algoritmos Genéticos são técnicas computacionais estocásticas¹ que visam otimizar sistemas complexos, cuja natureza não-determinística permite variações no resultado final de acordo com as condições iniciais e aleatoriedades presentes em cada etapa do algoritmo. São algoritmos probabilísticos que fornecem um mecanismo de busca paralela e adaptativa baseado no princípio de sobrevivência dos mais aptos e na reprodução (PACHECO. M. A. C, 1999). Neste contexto, um AG opera por meio de um processo evolutivo, o qual se baseia em princípios inspirados na seleção natural e reprodução biológica, a fim de encontrar um conjunto de soluções otimizadas para problemas computacionais. Os AGs têm sido aplicados em vários problemas NP-difíceis² com altas taxas de precisão. Existem algumas áreas de aplicação nas quais os AGs foram aplicados com sucesso, como processamento de imagens, agendamento, previsões e projetos de redes, segurança da informação etc. (Katoch et al., 2021). A figura 1 apresenta um pseudocódigo para algoritmos genéticos.

Conforme o pseudocódigo, a entrada do AG é definida com um número preestabelecido de tamanho da população e o número máximo de iterações a serem executadas, que são utilizados como parâmetros para o mapeamento do espaço de busca da solução. A população é entendida como um conjunto de possíveis resultados para o problema e é representada pela estrutura de cromossomos, que são definidos pelo programador como um modelo de objeto de solução ao problema proposto. Esses cromossomos contêm informações que caracterizam cada indivíduo da população, e podem ser interpretados como possíveis resultados para o problema em questão. O início do algoritmo é marcado pela geração de uma população inicial de cromossomos de tamanho n , que pode ser escolhida pelo usuário ou gerada aleatoriamente de acordo com o cenário do problema, ao passo que o contador de iterações é inicializado com $t = 0$. Em seguida, é atribuído um valor de aptidão para cada cromossomo da população, que é calculado de acordo com as métricas definidas em uma função de aptidão, que quantifica a qualidade de uma solução potencial. A partir deste ponto, enquanto o critério de parada não for atendido, o algoritmo seleciona os cromossomos baseando-se na sua aptidão, de modo a escolher os indivíduos mais adaptados, seguindo o

¹ Um estudo interessante sobre algoritmos genéticos e otimização estocástica para um sistema adaptativo complexo pode ser encontrado em (TRIGUEIRO; DE FARIAS, 2013).

² De acordo com Garey e Johnson (1979), um problema é considerado NP-difícil se todas as instâncias do problema podem ser reduzidas a uma instância de um problema NP-completo em tempo polinomial.

raciocínio da seleção natural; aplica a operação de cruzamento e a operação de mutação em consonância com as métricas probabilísticas estabelecidas, e, objetivando a variabilidade genética da população, substitui a população antiga pela nova população gerada e incrementa o contador de interação em um. O critério de parada de um AG é encontrar um cromossomo com uma adaptação adequada ao problema ou atingir a t -ésima interação.

Entrada:

Tamanho da população, n
Número máximo de iterações, MAX

Saída:

Melhor solução global, Y_{bt}

Início

Gere uma população inicial de n cromossomos Y_i ($i=1,2, \dots, n$)
Defina o contador de iterações $t = 0$
Calcule o valor de aptidão de cada cromossomo

Enquanto ($t < MAX$)

Selecione um par de cromossomos da população inicial baseado na sua aptidão
Aplique a operação de cruzamento no par de cromossomos selecionados com a probabilidade de cruzamento
Aplique a operação de mutação nos descendentes com a probabilidade de mutação
Substitua a antiga população pela nova população gerada
Incremente o contador t em 1

fim enquanto

retorne a melhor solução, Y_{bt}

fim

Figura 1 - Pseudocódigo de um algoritmo genético clássico (adaptado de Katoch et al., 2021)

A nova população p é formada por l cromossomos da população p e por $n - l$ cromossomos da população p . O critério adotado para o nascimento dos l cromossomos da população p é o mesmo usado para a escolha dos cromossomos progenitores da população p , o qual é aplicado no sentido inverso para determinar os l cromossomos da população p que devem morrer. Por fim, é calculado o valor da adaptação dos cromossomos da população p recém construída (VIEIRA, 2003). Ou seja, a nova população descendente é construída preservando as melhores características dos indivíduos da geração anterior, o que significa que quanto maior a adaptação, maior a chance de reprodução (OBITKO, 1998). É esperado que, ao final da execução do algoritmo, o resultado seja a melhor solução global para o problema proposto.

As próximas seções deste capítulo fornecem um detalhamento maior sobre cada etapa de um AG.

2.1 REPRESENTAÇÃO DO CROMOSSOMO

Como dito anteriormente, um cromossomo é uma estrutura que representa uma solução potencial para um dado problema a ser resolvido com AG. Ele é a unidade fundamental no processo de evolução natural, que serve como um portador de informações genéticas. De acordo com Holland (1975), um cromossomo é uma sequência de *bits* de tamanho n , que armazena as características essenciais para o desenvolvimento do indivíduo, como ilustra a figura 2.

0	0	1	1	0	1	0	0
---	---	---	---	---	---	---	---

Figura 2 - Representação de um *array* de bits (Autor, 2023).

Definição 2.1 (*vetor de bits*): Um cromossomo é representado por $k^{\rightarrow} = (x_1, x_2, \dots, x_n)$, onde $k^{\rightarrow}$ é o vetor de n elementos e cada $x_i \in \{0, 1\}$, representa o valor do i -ésimo elemento do vetor.

Na codificação binária, cada bit representa as características da solução. Ele fornece implementação mais rápida de operadores de cruzamento e mutação. No entanto, a conversão em formato binário pode exigir um esforço adicional e a precisão do algoritmo pode depender da qualidade dessa conversão. O fluxo de bits é alterado de acordo com o problema. Além disso, a codificação binária pode não ser apropriada para alguns problemas de design de engenharia, devido às suas características específicas (Katoch et al., 2021).

A modelagem do cromossomo está ligada com o tipo de problema que ele visa representar. Ao longo do desenvolvimento da área de AG, foram propostos diferentes tipos de codificações, como árvores de expressão, que são utilizadas quando as variáveis do problema possuem uma estrutura hierárquica e precisam ser avaliadas para gerar solução (KOZA, 1992); de conjuntos, no qual cada gene representa a presença ou ausência das variáveis do problema em um conjunto finito de opções; codificação de *Real-valued*, que está relacionada a problemas de otimização de variáveis contínuas reais, como a representação de quantidades de dimensões ou massas, por exemplo (EIBEN; SMITH, 2015); de permutação, usada quando as variáveis do problema têm uma ordem lógica e cada gene representa uma posição em uma sequência de elementos (GOLDBERG, 1989), e outras representações que podem ser vistas em (HOLLAND, 1975; KIM; HAN, 2000). Portanto, cabe ao programador entender o domínio do problema e propor um modelo de cromossomo de precisão, visto que a escolha pode afetar significativamente o desempenho e a eficácia de um AG.

2.2 REPRESENTAÇÃO DA POPULAÇÃO

A população é representada por Holland (1975) como um vetor formado por n cromossomos. Se cada cromossomo é formado por m elementos, logo a população será formada por uma matriz de n linhas e m colunas. Uma representação de uma população de arrays binários pode ser vista na figura 3.

```
population = [
    [0, 0, 1, 1, 0, 1, 0, 0],
    [1, 0, 0, 1, 1, 0, 1, 1],
    [1, 1, 1, 0, 1, 0, 0, 0],
    [0, 1, 1, 1, 1, 1, 0, 1],
    [0, 1, 0, 0, 0, 1, 0, 0]
]
```

Figura 3 - População de cromossomos. A população é uma matriz de dimensões 5 por 8, armazenada na variável `population` (Autor, 2023).

Definição 2.2 (Vetor de população) - Uma população é representado por $p^{\rightarrow} = (k1, k2, \dots, kn)$, onde $p^{\rightarrow}$ é o vetor de população de n elementos e cada $ki \in \{k^{\rightarrow}\}$, de acordo com a definição 2.1.

A população inicial de um AG pode ser gerada aleatoriamente ou por meio de uma distribuição de probabilidade que define a frequência que valores genéticos aparecem na população. Para cada iteração do algoritmo, é criada uma nova geração de indivíduos, que nada mais é do que uma população formada pelos melhores indivíduos da geração anterior e os novos indivíduos gerados a partir da ação dos operadores genéticos de cruzamento e mutação (seções 2.5 e 2.6). Estes, por sua vez, possuem características herdadas dos indivíduos da geração anterior, mas que apresentam variações genéticas que podem conter soluções ainda melhores. Quanto ao tamanho da população, se houver poucos cromossomos, o AG terá poucas possibilidades de realizar cruzamentos e somente uma parte pequena do espaço de soluções será explorada. Por outro lado, se houver muitos cromossomos, os AG se tornam lentos (OBITKO, 1998). Além disso, uma população muito grande pode levar a uma redução na variabilidade genética, visto que muitos indivíduos semelhantes apresentam soluções semelhantes. De acordo com as observações de Goldberg (1989), o tamanho da população na faixa de 50 a 200 indivíduos pode ser adequado para muitos problemas. Portanto, é importante entender o contexto do problema e realizar experimentações a fim de encontrar uma faixa de população ideal para a aplicação.

2.3 FUNÇÃO DE ADAPTAÇÃO

A função de adaptação, também conhecida como aptidão, é uma função matemática que mapeia os indivíduos de uma população em um valor que representa o quão adequada é essa solução em relação ao problema. Essa função é usada para avaliar a qualidade relativa das soluções candidatas em relação umas às outras e para selecionar as soluções mais adaptadas para o processo de reprodução, na etapa de seleção (seção 2.4). O exemplo 1 ilustra o processo de adaptação de uma população.

Exemplo 1 (Adaptação de uma população): Se $p^{\rightarrow}$ é um vetor de população de três cromossomos, no qual cada cromossomo é um vetor binário com seis genes³ que representam compostos químicos e a função de adaptação $f(p_i)$ mapeia o número de moléculas orgânicas no i -ésimo cromossomo dessa população, então de acordo com a figura 4:

$$p^{\rightarrow} = [[1, 0, 0, 1, 0, 1], [1, 0, 0, 1, 1, 1], [1, 1, 1, 0, 1, 1]]$$

$$f(p_1) = 3, f(p_2) = 4, f(p_3) = 5$$

Figura 4 - Cálculo de adaptação em uma população. Os bits 1 e 0 representam a presença ou ausência de moléculas orgânicas, respectivamente (Autor, 2023).

Os cromossomos um, dois e três têm adaptação 3, 4 e 5, respectivamente.

Definição 2.3 (Função de adaptação) - Considere uma população $p^{\rightarrow} = (k_1, k_2, \dots, k_n)$, conforme aponta a definição 2.2. A função de adaptação $f(k_i)$ mapeia cada indivíduo k_i da população para um valor numérico que representa a qualidade da solução. Esta função pode ser definida como $f: P \rightarrow \mathbb{R}$, na qual P representa uma população de soluções candidatas e f uma função de adaptação que calcula para cada indivíduo x de P um número real $f(x)$, que representa a sua adaptação.

Holland (1975) destaca que a escolha da função de adaptação é uma etapa crucial no projeto de AGs e influencia diretamente na capacidade do algoritmo em encontrar soluções ótimas para o problema. Ele sugere que a função de adaptação deve ser escolhida com cuidado, de modo a refletir corretamente a qualidade da solução e permitir que o algoritmo explore o espaço de busca com eficiência. Além disso, enfatiza que a função de aptidão deve ser definida

³ Um gene é uma unidade básica de informação hereditária representada por símbolos discretos, como bits, que é organizada em sequências lineares de cromossomos.

de forma clara e objetiva, de modo a permitir que os indivíduos sejam comparados e classificados com precisão.

2.4 OPERADOR GENÉTICO DE SELEÇÃO

A etapa de seleção visa escolher as soluções para um problema de acordo com a sua aptidão. É nesta etapa que cada indivíduo pode ser considerado para a reprodução de uma geração futura ou não, na esperança que reproduzam indivíduos ainda melhores, a depender de quão adaptado esse cromossomo está em relação ao ambiente e em relação aos outros membros da sua população. Há diversos tipos de operadores genéticos de seleção que apresentam características diferentes para cada abordagem de AGs. As próximas seções deste capítulo discutem os métodos de seleção por roleta, seleção por classificação, seleção por torneio e seleção elitista.

2.4.1 MÉTODO DE SELEÇÃO POR ROLETA

A seleção por roleta, também conhecida como seleção proporcional à aptidão, foi proposta por Baker (1985) e baseia-se em um processo de escolha de indivíduos com probabilidade proporcional à sua aptidão. Os indivíduos mais aptos têm maior probabilidade de serem selecionados para reprodução, mas indivíduos menos aptos ainda têm alguma chance de serem selecionados (ver exemplo 2). A alusão feita a uma roleta remete a probabilidade associada a cada indivíduo; logo, quanto maior a sua aptidão, mais larga será a sua “fatia” na roleta, como mostra a figura 6. A probabilidade de seleção é calculada dividindo-se a aptidão do i -ésimo cromossomo da população pela soma da aptidão de todos os cromossomos da população, como descreve a relação: $P_i = \frac{f(x_i)}{\sum_{k=1}^n f(k)}$.

Um possível algoritmo para o método de seleção por roleta é descrito na figura 5.

início

Some todos os valores de aptidão da população e chame de S

repita n vezes:

Gere um número randômico entre $(0, S)$ e chame de r

Percorra a população somando os valores das adaptações do cromossomo e chame de

s . Quando $s \geq r$, pare e retorne o cromossomo atual.

Figura 5 - Pseudocódigo da seleção por roleta (Autor, 2023).

Este tipo de seleção funciona de modo que a roleta é girada n vezes, onde n é o número de indivíduos na população. A cada giro, o indivíduo sob o marcador da roleta é selecionado para fazer parte da próxima geração. Mitchell (1996) aponta que este método resulta no número esperado de descendentes para cada indivíduo, mas como as populações são normalmente pequenas em AGs, o número real de descendentes alocados para um indivíduo muitas vezes está longe de seu valor esperado. Isso quer dizer que, caso a maioria dos cromossomos tenha uma adaptação muito alta, a parcela que tiver uma adaptação baixa terá uma probabilidade mínima de ser selecionada, levando o algoritmo a uma rápida convergência da população e acarretando uma perda prematura da diversidade genética.

Método de seleção por roleta

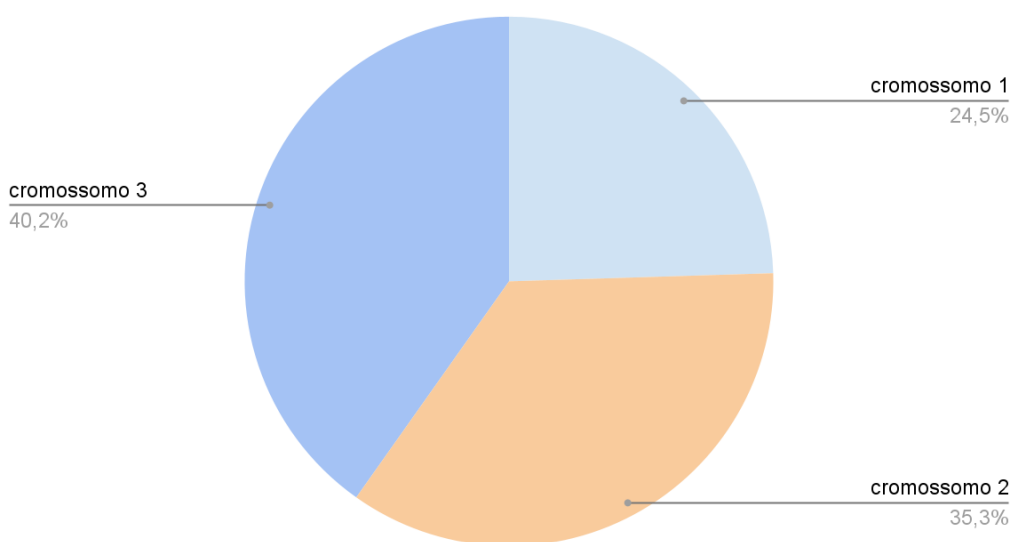


Figura 6 - Método de seleção por roleta. Cada cromossomo tem uma probabilidade que é proporcional a sua aptidão (Autor, 2023).

Exemplo 2 (Probabilidade de seleção): Partindo do Exemplo 1 da seção anterior, foi construída a tabela de probabilidades da população. A segunda e a terceira coluna representam os cromossomos de uma população e os seus respectivos valores de aptidão. Na terceira coluna, é atribuída a probabilidade de seleção correspondente a cada um dos cromossomos.

Tabela 1 - Probabilidade de seleção. A probabilidade associada a cada cromossomo foi encontrada por meio da divisão de cada valor de aptidão pela soma de todas as aptidões.

Indivíduo	Cromossomo	Valor de aptidão $f(x_i)$	Probabilidade $P(i)$
1	1 0 0 1 0 1	3	24,5 %
2	1 0 0 1 1 1	4	35,30%
3	1 1 1 0 1 1	5	40,2 %

2.4.2 MÉTODO DE SELEÇÃO POR CLASSIFICAÇÃO

De acordo com Goldberg e Deb (1991), a seleção por classificação é um método de seleção simples e eficaz que permite que indivíduos mais aptos sejam selecionados para reprodução, mas também que seja mantida alguma diversidade na população, permitindo que indivíduos menos aptos tenham a chance de serem selecionados. A seleção por classificação visa resolver o problema da dispersão que foi apresentado na seção anterior.

Este método se inicia por meio da avaliação da adaptação de cada cromossomo da população com base na $f(x_i)$. Então, os cromossomos são ordenados em ordem decrescente de aptidão e cada indivíduo recebe um *rank* com base na sua posição e classificação. Os ranks são normalizados para determinar as probabilidades de seleção para reprodução, com os indivíduos mais aptos tendo maior probabilidade de serem selecionados, mas mesmo os indivíduos menos aptos também têm chances de serem selecionados. A figura 7 representa o processo de seleção por classificação.

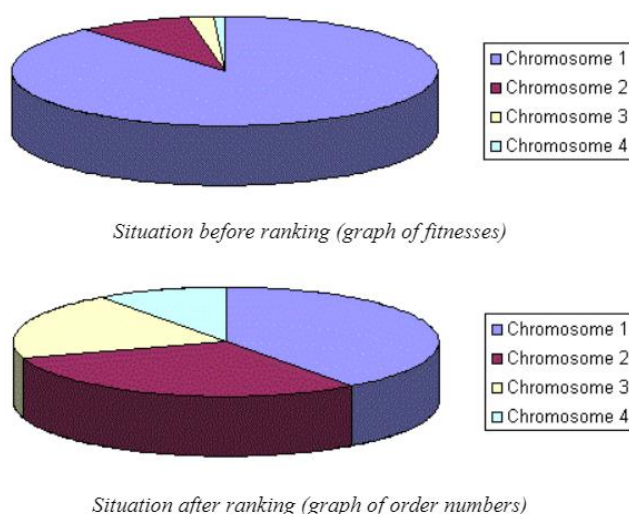


Figura 7 - Seleção por classificação. (Adaptado de OBITKO, 1998).

A imagem ilustra a situação dos cromossomos antes e depois do método de classificação. No primeiro gráfico, é apresentada a população de cromossomos de acordo com a sua aptidão, com uma situação de dispersão alta dos valores de aptidão. É perceptível que o cromossomo 4 (em azul claro) terá uma probabilidade diminuta de seleção, seguido dos cromossomos 3 (em amarelo) e o cromossomo 2 (no tom vinho) que também possuem baixa probabilidade de seleção quando comparados ao cromossomo 1 (em lilás), que representa a maior parte do gráfico e, conseqüentemente, domina a probabilidade de ser selecionado.

No segundo gráfico, o método de seleção por classificação tornou a situação dos cromossomos menos dispersa, já que os cromossomos foram ordenados de acordo com o *rank*. As probabilidades de escolha dos cromossomos 3 e 4 se tornaram mais competitivas em relação aos outros membros da população, que também sofreram mudanças significativas para se adaptar ao ambiente. Desse modo, as chances de formar uma geração mais diversificada e de evitar uma convergência prematura são maiores.

Em resumo, este tipo de seleção é simples e eficaz, o que permite selecionar indivíduos com base em sua aptidão em relação aos demais indivíduos da população. No entanto, pode ser computacionalmente intensivo, visto que requer a ordenação de toda a população com base em sua aptidão, o que pode ser impraticável para populações muito grandes em aplicações de larga escala.

2.4.3 MÉTODO DE SELEÇÃO POR TORNEIO

Os métodos de seleção por roleta e por classificação que foram descritos anteriormente requerem para cada nova população gerada o cálculo das operações de uma dada $f(x_i)$ e a P_i , e a ordenação de uma população inteira com base na *rank*, respectivamente. Apesar desses métodos de seleção serem eficazes em soluções com populações relativamente pequenas, a eficiência e o desempenho desses algoritmos podem estar comprometidos em populações grandes, visto que as operações citadas podem se tornar procedimentos demorados. De acordo com Mitchell (1996), a seleção por torneio é similar ao método de seleção por classificação, quanto à pressão de seleção, mas é computacionalmente mais eficiente e mais passível de implementar.

Este método baseia-se em realizar uma competição entre um número fixo de indivíduos escolhidos aleatoriamente na população, a fim de escolher o melhor deles para a reprodução e foi introduzida pela primeira vez por Holland (1975). A seleção por torneio é uma operação simples, que Mitchell (1996) descreve da seguinte maneira: Dois indivíduos são escolhidos ao

acaso da população. Um número aleatório r é então escolhido entre 0 e 1. Se $r < k$ (onde k é um parâmetro, por exemplo 0,75), o ajustador dos dois indivíduos é selecionado para ser um pai; caso contrário, o indivíduo menos apto é selecionado. Os dois são então devolvidos à população original e podem ser selecionados novamente.

O tamanho da população do torneio é um parâmetro importante que afeta o desempenho do AG. O tamanho refere-se ao número de indivíduos que participam da competição para serem selecionados como pais. Um tamanho de torneio grande pode levar a uma seleção mais diversificada de pais, mas também pode aumentar o tempo de execução do AG. Por outro lado, um tamanho de torneio pequeno pode levar a uma convergência prematura do AG, ou seja, a solução ótima é encontrada antes do término da execução do algoritmo. Logo, é importante realizar testes e adequar o tamanho da população de acordo com o problema em questão.

Então, a seleção por torneio é um método eficaz para utilizar em um AG, visto que é de simples implementação e é eficiente em termos de execução. Além disso, permite que os indivíduos de uma população tenham chances de participar da etapa de torneio, considerando que este método de seleção pode ser visto como uma maneira de aplicar a seleção natural em algoritmos genéticos, em que apenas os indivíduos mais aptos sobrevivem, como aponta Goldberg (1989).

2.4.4 MÉTODO DE SELEÇÃO ELITISTA

O conceito de seleção elitista já foi introduzido anteriormente, quando foi discutida a ideia de preservação dos melhores indivíduos de uma população para cada geração. Este método garante que as soluções com as melhores aptidões encontradas até o momento não sejam perdidas ao longo do processo evolutivo. Essa técnica pode ser realizada utilizando a substituição das soluções que possuem uma baixa aptidão pelas melhores ou criar uma população separada contendo apenas as soluções potenciais do problema, por exemplo. Obitko (1998) aponta que a abordagem elitista pode acelerar a performance de um AG porque previne a perda da melhor solução encontrada. Isto significa que o AG pode ter uma convergência mais rápida e eficiente para soluções ótimas; por outro lado, se a população for muito pequena, este método pode ter uma rápida convergência para soluções subótimas ou até mesmo mínimos locais. Além disso, pode aumentar a probabilidade de ocorrência de fenômenos como o "efeito de fim de festa", no qual a população converge prematuramente para uma solução ótima, antes de explorar adequadamente o espaço de busca.

Dessa forma, a abordagem elitista pode ser combinada com as outras técnicas de seleção, como as seleções por roleta ou torneio, para aumentar ainda mais a diversidade da população e evitar a estagnação em mínimos locais. Essa técnica também pode ser usada em problemas de otimização multiobjetivo, permitindo a preservação de soluções de alta qualidade para cada objetivo.

2.5 OPERADOR GENÉTICO DE CRUZAMENTO

A operação de cruzamento, também conhecida como *crossover*, é utilizada para gerar novas populações a partir da combinação de informações que estão presentes em dois cromossomos pré-existentes, que ocorrem com uma frequência de probabilidade p_c , simulando a recombinação de material genético que ocorre em uma reprodução sexuada. Essa operação foi introduzida por Holland (1975). A figura 8 ilustra o processo de cruzamento entre dois cromossomos.

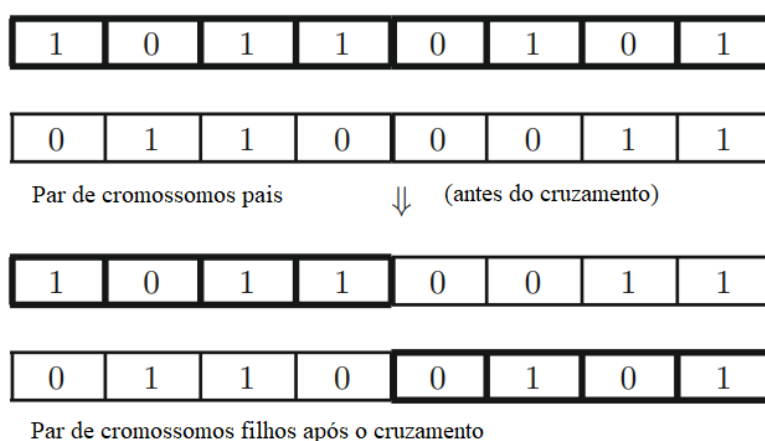


Figura 8 - Operação de cruzamento (Adaptado de Yang, 2014).

De acordo com a figura, os dois cromossomos pais trocam material genético, de modo que produzem dois cromossomos filhos, cada um com um filamento de informação que foi preservado dos seus pais. Isto significa que duas novas soluções foram geradas, preservando as características de ambas as soluções originais. O resultado é uma população de soluções com maior diversidade genética, o que pode favorecer a convergência para soluções mais ótimas (GOLDBERG, 1989).

As operações de cruzamento possuem os pontos de corte, que é o local onde ocorre a extração do material genético dos pais. Na figura 8, foi gerado apenas um ponto de corte na posição 3. O ponto corte, ou ponto de crossover, pode ser pré-definido ou gerado

aleatoriamente, e, inclusive, um cromossomo pode ter múltiplos pontos de corte em AGs para aumentar a eficiência evolucionária dos algoritmos (Yang, 2014). A figura 9 ilustra o processo de ponte de corte duplo.

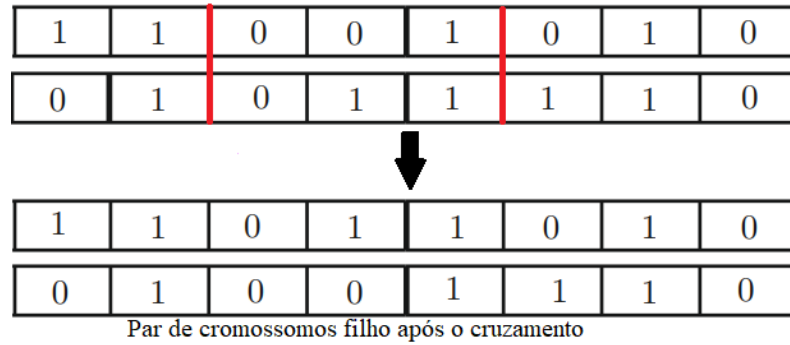


Figura 9 - Duplo ponto de corte. Nos dois primeiros cromossomos, os traços em vermelho são os pontos de corte (Autor, 2023).

É importante mencionar que a operação de cruzamento pode quebrar algumas sequências importantes de informação e, devido a isso, gerar indivíduos que não são totalmente adequados para o problema em questão. Porém, é possível controlar a taxa de cruzamento e ajustá-la para minimizar a possibilidade de danos ao material genético, utilizando a probabilidade de cruzamento p_c (seção 2.8).

2.6 OPERADOR GENÉTICO DE MUTAÇÃO

A operação de mutação visa introduzir variações aleatórias em soluções candidatas, com o intuito de aumentar a diversidade genética e evitar a convergência prematura para ótimos locais em uma população. Elas são essenciais para a evolução das soluções em um AG e são realizadas após o processo de cruzamento, conforme a probabilidade de mutação, p_m (seção 2.8). A seguir, serão discutidos os métodos de mutação por inversão de bit, mutação por inserção, mutação por troca e mutação uniforme.

2.6.1 MUTAÇÃO POR INVERSÃO DE BIT

A mutação por inversão de bit é uma operação simples de ser implementada computacionalmente. O objetivo deste método é gerar pequenas anomalias em um gene aleatório de um cromossomo, apenas fazendo a inversão de um bit 0 pelo bit 1 e vice-versa, conforme a imagem 10.

Definição 2.6.1 (Operação de inversão) - Dado um número i no intervalo $[0, t - 1]$, que representa a i -ésima posição de g , e b é o bit que ocupa essa posição, então:

$$g_i = \begin{cases} 1, & \text{se } b = 0 \\ 0, & \text{se } b = 1 \end{cases}$$

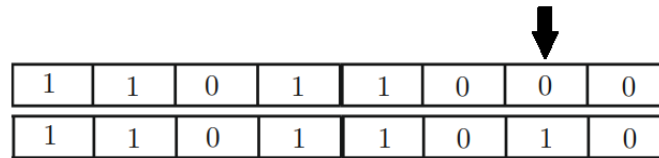


Figura 10 -Mutação por inversão de bit. A sexta posição do cromossomo que possui o bit 0 foi trocada pelo bit 1 (Autor, 2023).

2.6.2 MUTAÇÃO POR INSERÇÃO

A operação de mutação por inserção visa adicionar um novo gene a um determinado indivíduo da população, alterando o seu material genético. Ela permite a criação de novos indivíduos que não existiam na população anterior e pode ser vista como uma forma de aumentar a diversidade de uma população. Esta operação funciona adicionando um novo gene aleatório em uma determinada posição do cromossomo, que é selecionada aleatoriamente no intervalo $[0, t - 1]$. Após a inserção, todos os genes subsequentes são realocados para acomodar o novo gene, como ilustra a figura 11.



Figura 11 - Mutação por inserção (Adaptado de Kato et al, 2021).

De acordo com a figura, o gene 1 é selecionado aleatoriamente, antes da etapa de mutação. Em seguida, o gene 1 é realocado para a posição 4 do vetor, ao lado do gene 2, enquanto que os outros genes são realocados para as posições subsequentes. O resultado é um indivíduo diferente na população. Kato et al (2021) afirma que essa abordagem não modifica muito a ordem em que os genes aparecem e é utilizada em problemas de permutação.

2.6.3 MUTAÇÃO POR TROCA

A operação de mutação por troca envolve a troca aleatória de dois genes em uma solução de uma população, podendo ocorrer em qualquer posição da sequência genética do indivíduo. Este método de mutação acontece por meio da escolha de duas posições aleatórias dentro do intervalo $[0, t - 1]$. Em seguida, é feita a permutação entre os genes, gerando uma solução mutada que não havia antes na população. De acordo com Goldberg e Deb (1991), a mutação por troca é uma operação de mutação simples que pode melhorar significativamente a diversidade genética na população e, portanto, aumentar a chance de encontrar soluções ótimas. Como exemplo, tome um indivíduo que tem uma sequência genética representada por "ABCDEF"; a mutação por troca pode escolher aleatoriamente dois genes, digamos "A" e "F", e trocá-los de posição, resultando em "FBCDEA", ou seja, um novo indivíduo na população que pode ser uma solução potencial para o problema.

2.6.4 MUTAÇÃO UNIFORME

A operação de mutação uniforme consiste em alterar a informação genética de um gene g , escolhido aleatoriamente no intervalo $[0, t - 1]$. Os valores que g pode assumir estão contidos em um conjunto φ .



Figura 12 - Mutação uniforme (Adaptado de Kato et al, 2021).

De acordo com a figura 12, o indivíduo é um vetor de seis genes, no qual φ é o conjunto formado pelos bits 0 e 1. Dessa forma, o gene que ocupa a terceira posição foi escolhido aleatoriamente no ponto de mutação e teve o seu valor mudado para 1, escolhido aleatoriamente no conjunto φ . Kato et al (2021) afirma que esse tipo de mutação é usado para representação de indivíduos do tipo real e inteiro.

2.7 OPERADOR GENÉTICO DE SUBSTITUIÇÃO

Após as etapas de seleção, cruzamento e mutação, é o momento de construir uma nova população a partir dos melhores cromossomos que foram obtidos nas etapas anteriores, conforme a abordagem elitista. De acordo com Vieira (2003), a operação de substituição é uma operação de concatenação vetorial⁴, que recebe as populações de p , $\hat{p}$ e o parâmetro l , e retorna uma nova população formada pelos l cromossomos mais adaptados da população $\hat{p}$ e pelos $n - l$ cromossomos mais adaptados da população p . Isso significa que, ao fim da concatenação entre os vetores das populações citadas, o resultado é uma geração de indivíduos de elite e que possuem soluções potenciais para o problema, ou seja, com as melhores características e informações em seu material genético.

2.8 PROBABILIDADE DE CRUZAMENTO E MUTAÇÃO

As escolhas das probabilidades de cruzamento e mutação são fatores essenciais para o desempenho de um AG. Segundo Vieira (2003), a frequência que estes operadores são aplicados sobre os cromossomos de uma população pode afetar significativamente um AG, de modo a acelerar ou retardar a sua convergência.

Yang (2014) defende que a probabilidade de cruzamento é geralmente alta, de modo que $p_c = 0,7 \sim 1,0$. Por outro lado, a probabilidade de mutação é geralmente pequena, tal que $p_m = 0,001 \sim 0,05$. De maneira similar, Obitko (1998) defende que a taxa de cruzamento é alta, ficando em torno de $p_c = 0,8 \sim 0,95$, mas que para alguns tipos de problema, uma $p_c \approx 0,6$ atrai os melhores resultados; Para a taxa de mutação, o autor afirma que deve ser muito baixa, tal que as melhores faixas estão compreendidas em uma $p_m = 0,005 \sim 0,01$.

As análises da Yang (2014) apontam que, caso a p_c for muito pequena, o cruzamento ocorrerá de forma dispersa, o que não é eficiente para a evolução e pode não gerar descendentes adaptados. Por outro lado, se a p_m for muito alta, o AG pode “pular” soluções, mesmo que uma solução ótima esteja se aproximando. A escolha desses parâmetros é um fator crucial para o sucesso de um AG e se não forem escolhidos adequadamente, o AG pode fornecer um grande número de soluções inválidas ou indivíduos monstros, além de aumentar o tempo para a convergência de uma solução ótima; logo, uma p_c e p_m ajustadas de acordo com o problema

⁴ Uma operação de concatenação entre vetores pode ser entendida como uma operação de união entre conjuntos distintos. Uma definição mais robusta para essa operação pode ser encontrada em Vieira (2003), p. 26.

permitem ao AG explorar o espaço de buscas e manter a diversidade genética da população sem comprometer a convergência para a solução.

3. TRABALHOS RELACIONADOS

Um dos problemas clássicos da computação é encontrar o menor caminho em um grafo. A eficiência desse cálculo é crucial em várias áreas do conhecimento, como em sistemas de navegação, roteamento de redes de computadores, otimização de fluxo em sistemas de transporte, entre outras. Reduzir as distâncias do contexto do problema significa otimizar diretamente as variáveis que estão envolvidas nele, como o custo, o tempo, a velocidade de processamento e a latência, por exemplo. Além disso, é importante que o algoritmo escolhido para a resolução do problema seja eficiente, robusto e que possa processar as informações em um curto espaço de tempo, gerando informações consistentes para a economia do sistema como um todo.

Assim, este capítulo é dedicado à análise de alguns trabalhos que se assemelham com o tema desta monografia. As seções seguintes discutem brevemente os problemas clássicos da computação que abordam a otimização de caminhos, como o Problema do Caixeiro Viajante (TSP), o Problema de Roteamento de Veículos (VRP) e os softwares de otimização de recursos, o Recozimento Simulado, o Enxame de Partículas, bem como os respectivos trabalhos que serviram como base para o desenvolvimento da teoria por trás do modelo (capítulo 4). Para encontrar estes trabalhos, foi utilizado o mecanismo de pesquisa Google Scholar, que permite encontrar trabalhos científicos que estão indexados em vários repositórios, organizações acadêmicas e editoras, como é o caso do IEEE, Elsevier, Springer, Nature, MIT Press, Oxford Press etc. A busca foi feita utilizando os termos “*Genetic algorithms applied to Traveller Salesman problem*”, “*Path optimization using GA*”, “*Logistic optimization using GA*”, “*Vehicle routing problem using GA*”. A busca foi motivada pelo entendimento dos problemas clássicos de redução de distâncias que são tratados por meio de AG.

3.1 O PROBLEMA DO CAIXEIRO VIAJANTE

O problema do caixeiro viajante (TSP) consiste em um problema de otimização combinatória de classe NP-Completo, que visa encontrar a rota com menor custo em um conjunto de n cidades representadas em um grafo, de tal modo que o caixeiro visite todas essas cidades apenas uma vez, sem repetir o caminho e retorne à cidade inicial, conforme um caminho

Hamiltoniano⁵. Apesar de possuir um enunciado simplório, a complexidade computacional envolvida é intensa, visto que o número de rotas cresce em uma escala fatorial em função do número n de cidades, de modo que encontrar a sua solução em tempo polinomial ainda é uma tarefa em aberto na computação e matemática.

O artigo “*Genetic algorithms for the travelling salesman problem*” de Potvin (1996), discute a aplicação de AG para resolver o problema TSP. Neste trabalho, o autor faz uma breve passagem sobre as heurísticas presentes no campo de Pesquisa Operacional que foram utilizadas para resolver o TSP. Em seguida, apresenta uma revisão de literatura sobre AGs, mostrando como as estruturas e operações são representadas, discutindo os conceitos que são necessários para o entendimento da abordagem de AGs para o TSP. O autor também faz uma passagem nas abordagens de AGs que foram utilizadas para resolver o TSP na literatura e faz comparativos com o resultado dos seus experimentos, destacando a eficácia de AGs para resolver o TSP e abordagens de otimização combinatória semelhantes. Outro artigo interessante que discute a otimização de distância é “*Path optimization using genetic algorithm evolution*” de Low et al (2010). O artigo discute a otimização do caminho percorrido pelos braços robóticos que são utilizados em grandes empresas, usando AGs. Este problema pode ser resumido ao TSP, visto que a distância percorrida pelo braço é representada pelo plano de duas coordenadas, x e y . Além disso, os autores discutem a proposta da função objetivo e todas as etapas do domínio do problema com AGs, e mostram os resultados do AG que foi desenvolvido.

3.2 PROBLEMA DE ROTEAMENTO DE VEÍCULOS

Outro problema de otimização combinatória que possui uma grande repercussão no meio acadêmico e que se assemelha em alguns aspectos com o tema desta monografia é o Problema de Roteamento de Veículos. O VRP é um problema de otimização logística, no qual o seu objetivo é determinar a melhor rota para uma frota de n veículos que devem atender a um número x de clientes, de modo a minimizar o custo total de transporte. De modo geral, o VRP possui algumas variáveis de restrições, como restrições de horário, capacidade do veículo e horário de funcionamento de um cliente, que são abordadas de acordo com o contexto que visa responder. Além disso, cada problema possui custos associados às operações, como o combustível dos caminhões, o tempo em cada viagem, a distância percorrida etc. Logo, a função

⁵Um caminho Hamiltoniano em um grafo g é um caminho que visita cada nó exatamente uma vez, exceto pelo nó inicial que também é o nó final.

objetivo é encontrar um conjunto de rotas que atenda a todos clientes, utilizando a menor quantidade de veículos possível, abrangendo todas as restrições do problema, e minimizar o custo de operação.

O artigo “*Logistics Distribution Route Optimization Based on Genetic Algorithm*” de Xin et al (2022) é bastante interessante para a problemática envolvida nesta monografia. Neste trabalho, é discutido um estudo sobre a aplicação de AGs para a otimização de distribuição logística, no qual minimizar o tempo e o custo de entregas de mercadorias fazem parte do objetivo principal. Ao longo do artigo são feitas revisões de literatura sobre o problema de VRP, AGs, e conceitos sobre otimização logística para o modelo proposto. Por fim, o trabalho apresenta os resultados da simulação do modelo proposto do artigo, utilizando AGs para resolver o problema da otimização de distribuição, discutindo a sua eficácia no meio logístico.

Existem alguns softwares que permitem resolver o VRP para empresas, como é o caso do *onfleet*, uma plataforma que oferece entre as funcionalidades principais, o serviço de otimização das rotas de entregas, que surgiu como uma proposta interessante para a otimização de custos. Devido aos recursos inovadores que são oferecidos pela plataforma, como a prova de uma entrega e análise em tempo real, a plataforma possui um alto custo para uso, sendo que o plano mais básico custa \$500,00 dólares mensais e o seu plano ideal para operações de entrega custa \$1150,00 dólares mensais. Outra opção de software que fornece uma proposta similar ao *onfleet* é o *route4me*. Uma das principais funções desse software é otimizar rotas de entrega com alguns recursos adicionais, como evitar engarrafamentos e personalizar recursos diferentes para cada rota. Assim, a plataforma possui custo mensal para uso iniciando em \$499,00 dólares e seu plano completo custa \$899,00 dólares. Além disso, as plataformas não disponibilizaram informações sobre qual heurística de otimização foi implantada para o desenvolvimento dos softwares ou qual modelo de algoritmo utilizado, mas, são ótimas soluções que estão presentes no mercado atualmente.

3.3 O RECOZIMENTO SIMULADO

O Recozimento Simulado, também conhecido como *Simulate Annealing*, é uma meta-heurística que busca encontrar uma solução satisfatória para um problema complexo, simulando o processo do recozimento de um metal. De acordo com Bertsimas e Tsitsiklis (1993), o algoritmo de recozimento simulado é baseado na emulação do processo físico de resfriamento

de um sólido, em que se busca atingir uma configuração de energia mínima quando a estrutura é eventualmente congelada após um resfriamento gradual. O algoritmo inicia com uma solução inicial, permitindo que a sua energia seja reduzida de forma gradativa, de modo que a solução se move para uma solução vizinha com uma probabilidade determinada pela diferença de energia entre as soluções. Repete-se esse processo até que uma solução satisfatória seja alcançada ou quando o tempo limite seja atingido. Este algoritmo pode ser utilizado em problemas que possuem muitos mínimos locais, porque permite ao algoritmo alcançar soluções potencialmente melhores e os seus parâmetros devem ser ajustados de acordo com o contexto do problema.

3.4 O ENXAME DE PARTÍCULAS

O algoritmo do Enxame de Partículas (PSO) é uma heurística que busca encontrar a melhor solução em um problema de otimização, simulando o comportamento de um enxame de partículas que se movem com velocidades e posições distintas em um espaço multidimensional, baseado no comportamento social de um bando de pássaros. O objetivo deste algoritmo é buscar a solução ótima em um espaço de busca através da troca de informações entre indivíduos de uma população determinando qual trajetória cada um deles deverá tomar no espaço de busca (PACHECO, 2016). Este algoritmo é simples, de forma que a cada iteração as partículas se movem pelo espaço de busca na direção da melhor solução encontrada pelo enxame, balanceando a exploração do espaço de busca com a exploração das soluções próximas à posição da partícula. O algoritmo tem seu fim quando o critério de parada é atingido e a solução otimizada para o problema é encontrada. Essa abordagem pode ser utilizada em problemas cujos espaços de busca são amplos e possuem muitas variáveis, como otimização de funções matemáticas; otimização de sistemas complexos, como um sistema de distribuição de energia; e problemas de roteamento, como o TSP.

3.5 CONSIDERAÇÕES

Os algoritmos discutidos nesta seção se mostram ferramentas potenciais de otimização adequadas para a otimização do problema do caminho ótimo, visto que possuem a capacidade de buscar soluções em espaços de busca de forma eficiente. A escolha de AGs para a otimização do custo neste trabalho justificou-se pela sua especificidade de buscar soluções por meio de processos adaptativos e evolutivos de forma robusta em problemas combinatórios que envolvem amplos espaços de busca de forma rápida e simples, como foi visto no capítulo 2.

4. O MODELO

Neste capítulo, é abordado o desenvolvimento de um modelo de AG que simula a otimização da fase de transporte logístico de um pequeno empreendedor (PE). O objetivo do modelo é fornecer rotas de entrega mais seguras e minimizar as despesas associadas a essa fase, resultando em uma redução orçamentária total do PE para a realização da distribuição dos seus insumos em um determinado número de locais. Para isso, as definições utilizadas na formalização do algoritmo genético são detalhadamente descritas nas seções seguintes, em conjunto com a análise dos resultados obtidos e a contextualização dos mesmos.

4.1 DEFINIÇÕES TÉCNICAS DO MODELO

Para o desenvolvimento do modelo, a linguagem escolhida foi Python, devido a prática do autor e a disposição de bibliotecas científicas, como o NumPy, pandas e matplotlib, que fornecem funcionalidades poderosas para a modelagem de aplicações científicas e recursos de visualização e análise de dados. De acordo com VanderPlas (2016), Python é uma das linguagens mais versáteis e poderosas para a análise de dados e a computação científica. Logo, como a linguagem oferece esses recursos, foi escolhida para a produção do modelo.

Por se tratar de um modelo de algoritmo genético, ou seja, uma aplicação científica, o ambiente de desenvolvimento escolhido foi o Google Colaboratory, também conhecido como Colab. Nele, é possível desenvolver e executar códigos escritos na linguagem Python, combinar recursos de texto, imagens e outros formatos diretamente no navegador e não necessita de nenhuma configuração adicional. Além disso, o Colab permite aproveitar os recursos de visualização de dados, dados aleatórios e bibliotecas que já estão integradas na plataforma, dispensando a instalação de recursos adicionais, o que o torna uma ferramenta ideal para o desenvolvimento de aplicações científicas de cunho acadêmico.

4.2 ESPECIFICAÇÕES DO MODELO GENÉTICO

Esta seção descreve o conjunto de especificações utilizadas para a formalização do modelo, que foram descritas na metodologia de desenvolvimento deste trabalho (seção 1.4). A partir deste ponto, serão apresentadas as características gerais abordadas para o domínio do problema, tais como a função objetivo, etapa de seleção, mutação, cruzamento e custo; mais adiante, serão apresentados os requisitos para as simulações e validações do modelo, utilizando diferentes taxas de probabilidade de mutação, tamanho da população, número de gerações,

número de pontos e tamanho da população de elite a fim de entender o comportamento do AG e contextualizar a sua aplicação para a fase de transporte logístico do PE.

4.2.1 DEFINIÇÃO DO PROBLEMA

A definição do problema consiste em minimizar o orçamento total associado à fase de transporte logístico de um PE local que distribui os seus insumos para uma quantidade n de clientes diariamente ao redor da cidade, utilizando apenas um carro para realizar as entregas. A única restrição do problema consiste em realizar todas as entregas do dia, ou seja, passar em todos os pontos os quais residem os seus clientes e voltar ao ponto inicial. Além disso, a ordem de entregas não é um problema, uma vez que todos os clientes que compram os seus produtos, os recebe no mesmo dia de compra, considerando que todos os pedidos são realizados na parte da manhã e o conjunto de entregas do dia é feito pela tarde, em um único percurso, uma única vez ao dia. As variáveis envolvidas no problema são: distância e a soma de custo da etapa de distribuição (seção 4.2.3), diretamente proporcionais entre si. Desse modo, pretende-se modelar um AG capaz de reduzir o orçamento financeiro total para realizar a fase de transporte de insumos de maneira otimizada, com o objetivo de baratear os custos de operação e instruir rotas mais seguras e eficientes para o PE.

4.2.2 REPRESENTAÇÃO DE SOLUÇÕES

A representação de soluções deste modelo é uma adaptação da representação proposta por Stoltz (2018), na qual utilizou a biblioteca matplotlib da linguagem Python para ilustrar a evolução da solução de AGs. Desse modo, foi possível visualizar os recursos gráficos da evolução das potenciais soluções desenvolvidas ao longo das gerações das populações e as rotas a serem seguidas, gerando informações para a contextualização e análise dos dados.

4.2.3 REPRESENTAÇÃO DO CUSTO

A representação do custo do modelo seguiu a definição do custo da fase de transporte de produtos, que foi definido por Borba e Gibbon (2012). Segundo os autores, o custo de transporte de entrega próprio assume todo o gasto necessário para atender o pedido para cada cliente. Nesse cálculo, são consideradas todas as despesas com o frete, tais como: material, combustível, equipamento, manutenção do transporte, seguro etc. Para o modelo proposto neste trabalho, foi simulado a geração de custo associada a cada localização de forma aleatória, com intuito de se aproximar ainda mais do cenário real.

4.2.4 FUNÇÃO OBJETIVO

Dados a relação de custo associado a uma rota, $c = d \cdot s$, nos quais d e s representam a distância e a soma dos recursos que estão envolvidos para realizar essa rota, respectivamente; $p = \{p_1, \dots, p_n\}$ o conjunto de n pontos de entrega, $r = \{r_1, \dots, r_n\}$ o conjunto de todas as rotas formadas por p e $f(c_1, \dots, c_n)$ são todos os custos relacionados a r , então:

Deseja-se minimizar: $f(c_1, \dots, c_n)$

Satisfazendo: $g(c_1, \dots, c_n) = k$

em que:

$f(c_1, \dots, c_n) = \sum_{i=1}^n (d_i \cdot s_i) \rightarrow$ função objetivo formada pelas variáveis do modelo (custo total)

$g(c_1, \dots, c_n) = k \rightarrow$ restrição: encontrar a rota k de menor custo que passe por todos os pontos e retorne ao ponto inicial, no qual g mapeia os custos associados ao conjunto r .

Dessa forma, como as variáveis do modelo são diretamente proporcionais, minimizar a distância total percorrida significa, portanto, otimizar os custos totais associados à etapa de transporte de mercadorias.

4.2.5 OPERADORES GENÉTICOS

O processo evolutivo deste modelo segue as etapas que foram descritas no capítulo 2. Os operadores genéticos foram definidos da seguinte maneira:

1. Um **gene** é definido como um ponto cartesiano, que é representado por coordenadas (x, y, z) tal que $x, y, z \in \mathbb{R}$; de tal forma que as duas primeiras coordenadas são de localização e a terceira representa o somatório de custo de transporte para este ponto;
2. Um **cromossomo** é definido como uma rota, que é formada por um dado conjunto de pontos;
3. Uma **população** é definida como um conjunto de rotas;
4. A **função de adaptação** calcula a distância de uma rota por meio da distância euclidiana; posteriormente, para a etapa de classificação, é considerado a razão do inverso da distância para atribuir uma pontuação a aptidão da rota: quanto maior essa pontuação, menor será a rota;
5. Os **pais** são definidos como duas rotas que serão combinadas para formar os filhos;

6. O **cruzamento** é definido como a combinação ordenada do material genético proveniente dos pais, ou seja, a seleção aleatória das porções de uma rota para formar uma rota descendente, de modo a não repetir nenhum ponto;
7. A **mutação** é definida como a troca de posição entre dois pontos, de modo a introduzir novas rotas na população;
8. A abordagem de **seleção** é o elitismo, no qual as rotas que possuem as menores distâncias são as selecionadas por meio da seleção proporcional a aptidão.

4.2.6 CRITÉRIO DE PARADA

O critério de parada seguiu a abordagem tradicional de um AG que foi abordado no capítulo 2, no qual o AG tem seu fim quando atinge um determinado número de iterações, que, neste caso, é o número de gerações que o modelo deve evoluir para encontrar a solução ótima.

4.2.7 VALIDAÇÃO DO MODELO

A etapa de validação desempenhou um papel importante para a verificação do modelo, visto que, ao utilizar dados gerados aleatoriamente, foi possível simular diversas situações que podem ocorrer no mundo real. Nesse contexto, os pontos de entrega de insumos e o custo associado foram gerados de forma aleatória, visto que esses parâmetros influenciam diretamente no resultado das simulações. Além disso, a definição de taxa de mutação foi escolhida de forma cautelosa por meio de testes, até encontrar um intervalo ideal. No presente estudo, foram realizadas um total de 30 simulações com o objetivo de avaliar o comportamento do modelo e analisar o contexto dos resultados. Cada simulação foi executada com o conjunto de parâmetros escolhidos, como o número de pontos de entrega a serem gerados e o número de gerações que o modelo deve gerar, de modo a explorar diferentes cenários e configurações do AG, com cada custo associado a um ponto da rota gerado de forma aleatória. Por fim, os resultados foram analisados e descritos na seção 4.3, na qual os parâmetros escolhidos estão relatados em cada simulação, de modo a entender e contextualizar a efetividade do modelo proposto e a sua validade para a otimização do problema logístico do PE.

4.3 RESULTADOS E DISCUSSÕES

Neste capítulo são abordados os resultados e discussões da etapa de validação do modelo, que foi descrita na seção anterior. As simulações foram feitas com o intuito de explorar a capacidade e acurácia do modelo, de modo a entender o seu comportamento em diferentes

cenários. Os resultados das simulações estão descritos nas tabelas a seguir, nas quais os parâmetros envolvidos no processo evolutivo estão discriminados; também, são apresentados os resultados das rotas ilustradas e o gráfico do processo evolutivo que foram gerados pelo modelo evolucionário proposto.

4.3.1 PRIMEIRA ETAPA DE SIMULAÇÃO

A simulação descrita na tabela 2 mostra os resultados de um conjunto de cinco simulações iniciais. Para esta etapa, todos os parâmetros que foram descritos nas colunas dois a seis foram escolhidos para realizar os primeiros testes e verificar se o modelo gerou valores diferentes. Estes resultados apontam que o modelo está funcionando de acordo com o esperado e obtendo os valores de otimização, no qual os preços e as distâncias resultantes estão de acordo com a otimização obtida.

Tabela 2 - Simulações do modelo evolucionário. Foram feitas 5 simulações com parâmetros diversos, que são descritos em cada coluna.

SIMULAÇÃO	N. DE PONTOS	TAM. DA POP.	TAM. POP. ELITE	TAXA DE MUTAÇÃO	N. DE GERAÇÕES	DIST. INICIAL (km)	DIST. FINAL (km)	PREÇO (R\$)	OTIMIZAÇÃO (%)
1	10	50	10	0,01	100	162,75	63,91	35,06	60,73
2	15	50	12	0,01	150	220,33	110,03	58,3	50,06
3	20	80	15	0,009	200	309,94	112,16	61,83	63,81
4	25	100	20	0,008	250	417,08	169,81	106,56	59,29
5	30	200	25	0,01	300	436,46	145,66	92,15	66,63

A figura 13 ilustra o resultado da melhor rota encontrada da simulação 5, seguido do gráfico da evolução da população, que encontrou o ótimo global por volta da geração 110 em um conjunto de 300 gerações.

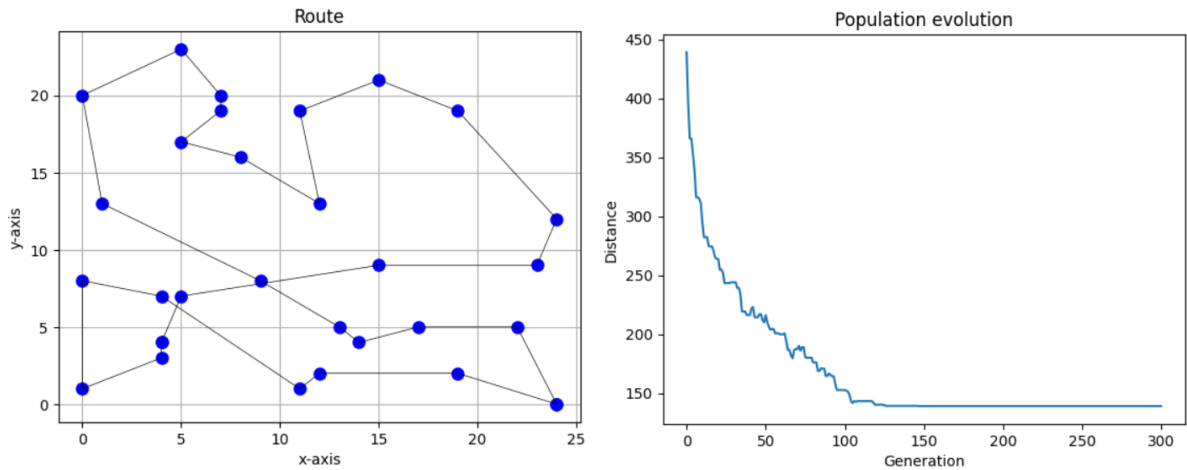


Figura 13 - Conjunto de gráficos da simulação 5 (Autor, 2023).

4.3.2 SEGUNDA ETAPA DE SIMULAÇÃO

Para esta etapa, foi considerado um total de 25 pontos para realizar todas as simulações, o que representa um pouco mais de 600 sextilhões de rotas diferentes.

Tabela 3 - Continuação das simulações do modelo evolucionário. Para essa rodada, foram considerados 25 pontos e diferentes valores para os parâmetros genéticos.

SIMULAÇÃO	N. DE PONTOS	TAM. DA POP.	TAM. POP. ELITE	TAXA DE MUTAÇÃO	N. DE GERAÇÕES	DIST. INICIAL (km)	DIST. FINAL (km)	PREÇO (R\$)	OTIMIZAÇÃO (%)
6	25	100	15	0,007	100	367,04	138,83	74,58	62,18
7	25	120	20	0,008	200	362,21	144,05	83,93	60,23
8	25	150	25	0,009	300	307,06	106,84	59,25	65,21
9	25	200	30	0,01	400	366,25	111,76	63,56	69,49
10	25	250	30	0,015	500	282,4	102,49	58,97	63,71

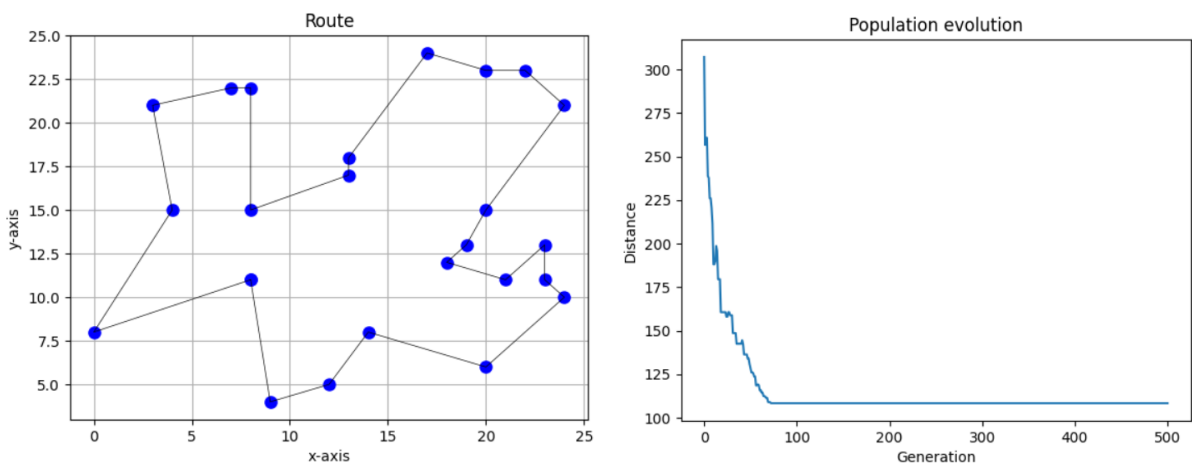


Figura 14 - Conjunto de gráficos da simulação 10 (Autor, 2023).

Os valores da população e da população de elite foram escolhidos gradualmente, assim como a mutação e o número de gerações, como aponta a tabela 3. O menor valor encontrado de otimização (linha 2) foi de aproximadamente 60%, enquanto o maior representou quase 70%. (linha 4) Essa diferença deveu-se ao fato da escolha da taxa de mutação, que foi menor do que os valores recomendados na seção 2.8. Além disso, um número maior de população representa um maior espaço de busca de soluções, o que mostra que houve mais rotas para considerar. Na figura 14, nota-se que a melhor rota foi encontrada antes da centésima geração e não encontrou uma rota menor durante as gerações seguintes. O gráfico ao lado ilustra essa rota.

4.3.3 TERCEIRA ETAPA DE SIMULAÇÃO

Nesta etapa, foi considerado um total de 30 pontos de localização, realizando mudanças no tamanho da população, no número de gerações e o aumento gradativo da probabilidade de mutação.

Tabela 4 - Continuação das simulações do modelo evolucionário. Para essa rodada, foram considerados 25 pontos e diferentes valores para os parâmetros genéticos.

SIMULAÇÃO	N. DE PONTOS	TAM. DA POP.	TAM. POP. ELITE	TAXA DE MUTAÇÃO	N. DE GERAÇÕES	DIST. INICIAL (km)	DIST. FINAL (km)	PREÇO (R\$)	OTIMIZAÇÃO (%)
11	30	100	20	0,01	500	192,81	64,47	38,46	66,56
12	30	150	20	0,02	550	224,28	82,71	50,17	63,12
13	30	200	20	0,03	600	207,11	126,4	88,06	38,97
14	30	300	20	0,04	650	239,39	139,26	89,87	41,83
15	30	350	20	0,05	700	249,65	138,92	83,68	44,35

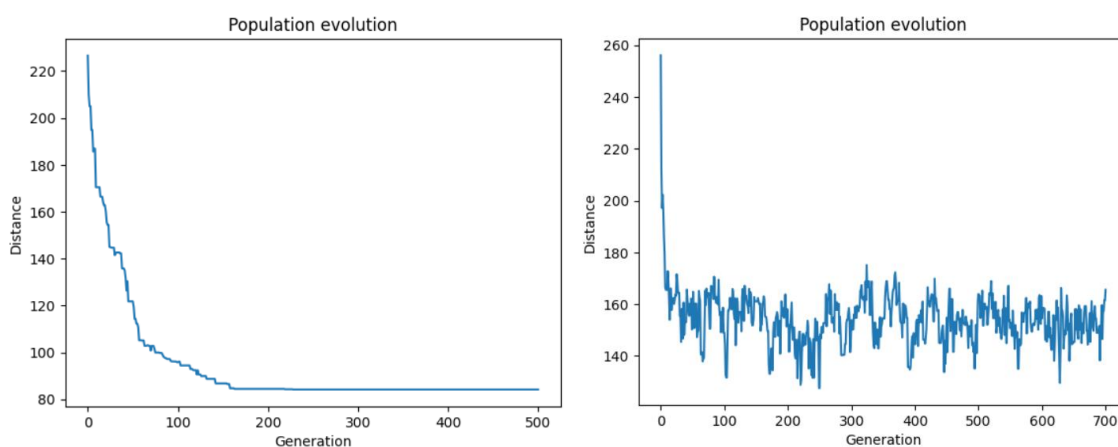


Figura 15 - Gráficos de evolução da população (Autor, 2023).

O aumento da probabilidade de mutação alterou o comportamento do modelo, como é possível ver nas linhas 3, 4 e 5 da tabela 4. A diminuição na eficiência ocorreu porque a probabilidade de mutação foi aumentada em 1% a cada simulação, o que fez com que as melhores rotas que foram encontradas sofressem mudanças significativas nas suas distâncias, como é possível notar na figura 15. O primeiro gráfico representa o comportamento da simulação 11, no qual a probabilidade de mutação foi respeitada e o modelo seguiu preservando as melhores soluções; já o segundo gráfico mostra o comportamento da simulação 15, no qual os ruídos representam uma frequência maior de mutação que as rotas sofreram de acordo com o passar das gerações. O aumento de 4% de probabilidade fez com que as rotas tivessem um desvio de 80 quilômetros para mais ou para menos nas distâncias das populações subsequentes.

4.3.4 QUARTA ETAPA DE SIMULAÇÃO

Nesta etapa, foi considerado um total de 50 pontos. Além disso, os parâmetros de população de elite e número de gerações foram modificados para verificar o que acontece quando se tem poucos membros para compor a população de elite.

Tabela 5 - Continuação das simulações do modelo evolucionário. Para essa rodada, foram considerados 50 pontos e a diminuição significativa do tamanho da população de elite.

SIMULAÇÃO	N. DE PONTOS	TAM. DA POP.	TAM. POP. ELITE	TAXA DE MUTAÇÃO	N. DE GERAÇÕES	DIST. INICIAL (km)	DIST. FINAL (km)	PREÇO (R\$)	OTIMIZAÇÃO (%)
16	50	50	10	0,005	1000	306,76	89,16	67,51	70,93
17	50	60	8	0,01	2000	304,75	101,15	70,04	66,81
18	50	70	6	0,015	3000	309,18	198,67	140,5	35,74
19	50	80	4	0,025	4000	298,71	213,44	160,16	25,55
20	50	90	2	0,04	5000	312,31	249,06	181,31	20,25

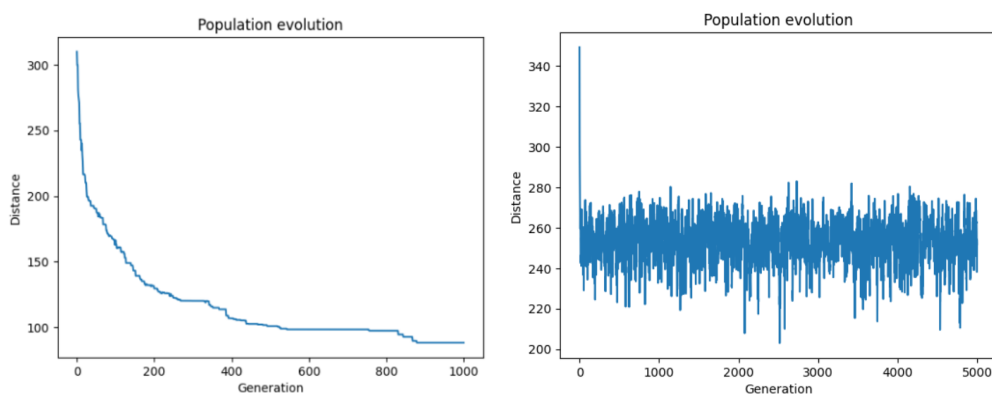


Figura 16 - Gráficos de evolução da população (Autor, 2023).

A diminuição da população de elite acarretou uma redução muito significativa do desempenho do modelo. Uma população de elite de 10 indivíduos resultou em uma otimização de aproximadamente 70% do custo ao longo de 1000 gerações, enquanto, para uma população de elite de 2 indivíduos a otimização foi de 20% ao longo de 5000 gerações (tabela 5). A figura 16 retrata esse cenário, no qual o gráfico da esquerda (simulação 16) tem o seu comportamento dentro da normalidade e encontra o seu ótimo global por volta da geração 820; diferente do gráfico da direita (simulação 20), que apresentou ruídos devido ao algoritmo possuir um número muito baixo de indivíduos de elite para selecionar e, desse modo, as rotas de maior distância foram selecionadas, resultando em gerações futuras com baixa qualidade nas soluções.

4.3.5 QUINTA ETAPA DE SIMULAÇÃO

Nesta fase, foi considerado um número de 25 locais para entrega. Os parâmetros de tamanho da população e tamanho da população de elite tiveram uma redução no seu valor para entender como o modelo se comporta no decorrer das gerações.

Tabela 6 - Continuação das simulações do modelo evolucionário. Para essa rodada, foram considerados 25 pontos e a diminuição significativa do tamanho das populações.

SIMULAÇÃO	N. DE PONTOS	TAM. DA POP.	TAM. POP. ELITE	TAXA DE MUTAÇÃO	N. DE GERAÇÕES	DIST. INICIAL (km)	DIST. FINAL (km)	PREÇO (R\$)	OTIMIZAÇÃO (%)
21	25	5	3	0,01	100	103,27	76,94	42,54	25,5
22	25	10	6	0,01	200	116,14	53,86	35,41	53,62
23	25	15	9	0,01	300	125,88	55,58	32,6	55,85
24	25	20	12	0,01	400	122,91	52,36	32,81	57,4
25	25	25	15	0,01	500	139,55	52,46	29,81	62,41

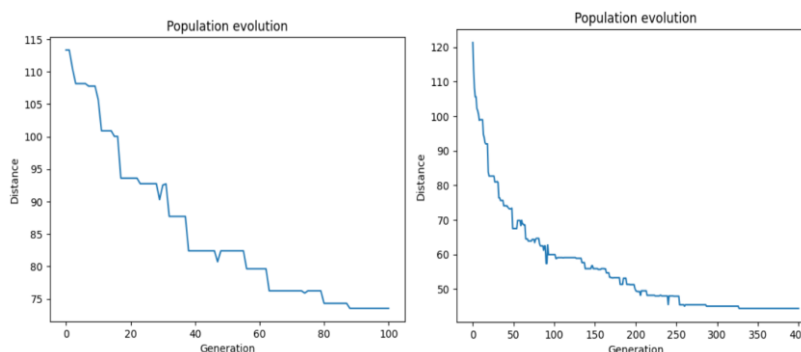


Figura 17 - Gráficos de evolução da população (Autor, 2023).

Os resultados apontados na tabela 6 ilustram a situação de um número muito baixo de população, quando comparados aos valores apresentados na segunda e terceira etapa de simulação. Para o tamanho da população de elite, buscou-se valores proporcionais ao tamanho do espaço de busca das soluções. Percebeu-se que, quanto menor o espaço de busca, menor é a eficiência do modelo, com apenas 25.5% de otimização (linha 1); além disso, quando o espaço de busca dobrou o seu valor, o modelo ofereceu uma redução de 53% do custo percorrendo 100 gerações a mais. E, finalmente, para o melhor resultado de otimização que o modelo atingiu, o algoritmo trabalhou com as maiores populações e um número de gerações alto, o que proporcionou um refinamento maior na busca da menor distância (linha 5). A figura 17 mostra os gráficos da simulação 21 e simulação 24, respectivamente.

4.3.6 SEXTA ETAPA DE SIMULAÇÃO

Para esta última etapa de simulação, utilizou-se a combinação dos parâmetros que obtiveram os maiores valores de otimização nas simulações anteriores, aplicados em um conjunto de 25 pontos.

Tabela 7 - Últimas simulações do modelo evolucionário. Foram escolhidos os parâmetros que geraram os melhores resultados.

SIMULAÇÃO	N. DE PONTOS	TAM. DA POP.	TAM. POP. ELITE	TAXA DE MUTAÇÃO	N. DE GERAÇÕES	DIST. INICIAL (km)	DIST. FINAL (km)	PREÇO (R\$)	OTIMIZAÇÃO (%)
26	25	200	25	0,01	500	138,96	47,85	29,3	65,57
27	25	200	30	0,01	400	142,63	49,93	28,91	64,99
28	25	150	20	0,02	550	136,22	46,19	30,04	66,09
29	25	50	10	0,005	1000	151,18	47,9	31,03	68,32
30	25	100	50	0,01	2000	168,7	42,46	24,24	74,83

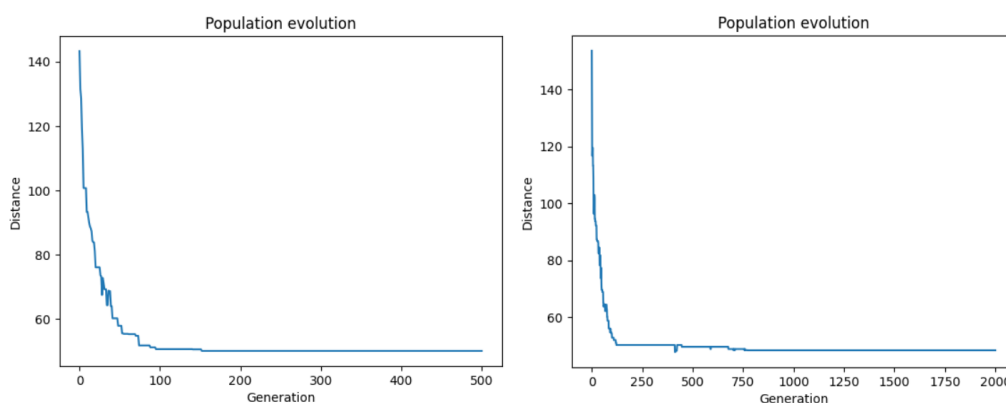


Figura 18 - Gráficos de evolução da população (Autor, 2023).

Nesta simulação, as recomendações dos parâmetros genéticos que foram discutidas no capítulo 2 foram utilizadas (tabela 7) com o intuito de se obter os melhores valores para a otimização das rotas e do custo, respectivamente. Foi possível constatar que uma população de 200 indivíduos, combinados com uma população de elite entre 20 e 50 indivíduos, utilizando uma taxa de mutação que varia entre 0.005 e 0.02, e com um número que varia entre 500 e 2000 gerações são suficientes para obter uma acurácia de mais de 70% neste modelo. O comportamento desses parâmetros é refletido na figura 17, na qual os gráficos são respectivamente das simulações 26 e 30. Nota-se que as curvas são mais sutis do que as curvas apresentadas nas outras simulações, devido a influência dos parâmetros escolhidos, que estão diretamente relacionados com a qualidade das soluções. Também, notou-se que os valores de preço⁶ que foram apresentados sofreram uma redução muito significativa, visto que o valor de preço final segue a otimização consequente que foi aplicada na rota. A figura 19 ilustra as diferenças entre uma rota otimizada mais segura de uma possível rota inicial de uma possível lista de clientes, antes da otimização, conforme a simulação 30 (linha 5).

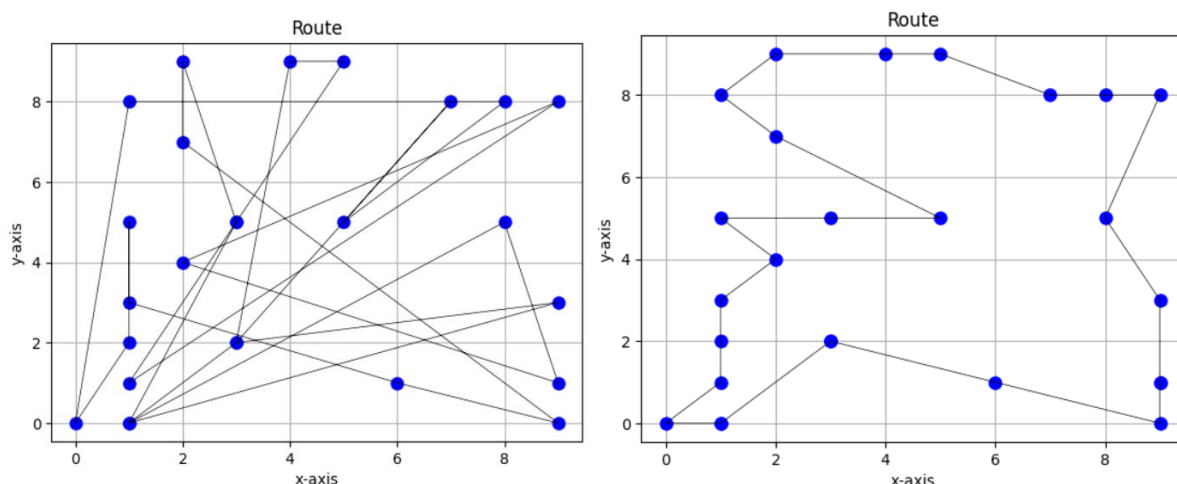


Figura 19 - Gráficos das rotas inicial e final, respectivamente (Autor, 2023).

Por fim, considera-se que o modelo apresentado neste trabalho é o esboço inicial de aplicação de grande valia para os pequenos empreendedores, visto que possibilita uma economia de recursos significativa para o transporte de mercadorias, já que esta etapa é a de maior custo no processo logístico, como visto anteriormente. Além disso, pode promover rotas mais eficientes e seguras, e ter uma maior disponibilidade do setor logístico da sua empresa.

⁶ O valor do preço inicial foi omitido devido ao tamanho limitado da tabela, mas foi abordado no modelo.

5. CONCLUSÃO

O presente trabalho propôs um modelo de algoritmo genético rápido, simples e eficaz para a redução dos recursos que são utilizados na fase de transporte logístico do pequeno empreendedor. O processo evolutivo que foi utilizado para o desenvolvimento do modelo utilizou os operadores genéticos de seleção proporcional a aptidão para preservar as melhores soluções no processo evolutivo; cruzamento, no qual as rotas foram combinadas para gerar um filho; a mutação por troca, para introduzir rotas inéditas em uma população e outros recursos; além disso, foram definidos os custos logísticos associados a cada ponto da rota, diretamente proporcionais à distância, simulados aleatoriamente para encontrar valores próximos dos valores reais, somados com o preço do combustível, que também está relacionado com a distância e são influenciados com a ordem de decisão da escolha dos pontos. Também, os recursos gráficos mostrados permitem interpretar o processo evolutivo e a evolução das rotas logísticas, com a distinção da melhor e a pior rota.

As contribuições deste trabalho permitiram uma visão sobre o barateamento de custos logísticos no cenário do pequeno empreendedor para o desenvolvimento de uma ferramenta de barateamento de custos no Brasil, tendo em vista as suas dificuldades de planejamento e organização na fase de distribuição de insumos. Esta proposta também pode ser aplicada para negócios de grande porte, uma vez que as empresas querem baratear o custo das suas operações e maximizar os lucros; reduzir o tempo de operação, tendo em vista que turnos mais longos de trabalho possuem despesas maiores com os seus funcionários; e ter um processo de distribuição logística integrada à sua empresa, uma vez que empresas terceirizadas podem ter custos maiores. Como foi discutido ao longo do texto, o modelo forneceu uma operação de custo que representou $\frac{1}{4}$ do valor total, o que fornece um cenário atrativo para negócios que estão em fase de planejamento, execução e até empresas que já estão consolidadas no mercado e desejam melhorar a sua fase de distribuição. Assim, o modelo proposto mostrou-se uma aplicação potencial para a melhoria do cenário logístico brasileiro, uma vez que contribui para o desenvolvimento de rotas eficazes para o transporte de insumos.

Existem algumas limitações envolvidas neste trabalho, como a estimativa do custo em cada ponto de distribuição e a determinação do tempo entre cada entrega, o número de mercadorias que o veículo pode transportar e quais vias ele pode utilizar em uma cidade. Estas informações são fatores determinantes para o sucesso de um percurso otimizado, uma vez que

todos os clientes devem receber todas as suas mercadorias no mesmo dia de entrega, o que pode acarretar uma ocupação total do espaço do veículo de distribuição e prejudicar a segurança do distribuidor, devido às restrições de tamanho e peso da carga que um veículo pode transportar. Estas restrições podem ser resolvidas por meio de estudos de caso aprofundados no comportamento dos consumidores dos produtos ofertados pelos empreendedores, em um contexto geral, uma vez que são oferecidos diferentes cenários de produtos e negócios.

Como trabalhos futuros, espera-se a integração deste modelo com um sistema de análise de tráfego, a fim de encontrar o melhor momento para a realização das rotas de distribuição e prevenir o desperdício de tempo em congestionamentos ou vias que possam estar em manutenção, por exemplo; também, a sua integração com sistemas de navegação para fornecer rotas precisas por meio de uma API que possa ser integrada em aplicações reais de otimização logística, por meio de softwares ou aplicativos de uso pessoal. Além disso, deve ser feito um estudo mais aprofundado dos custos para os pontos de entrega, a fim de otimizar com precisão as despesas do PE com a fretagem de mercadorias; também, entender o cenário de distribuição para determinar a quantidade de pacotes que o veículo pode transportar com segurança. Assim, pode ser possível o seu uso para os diferentes públicos, como as plataformas de transporte como o Uber, pessoas que desejam realizar entregas para pequenos negócios locais etc. Também, o modelo pode ser aperfeiçoado para a distribuição logística em pontos de difícil acesso, tais como comunidades, regiões remotas e tribos indígenas, visto que não são todos os tipos de veículos que podem trafegar em diferentes terrenos, como os rios, lagos, escadarias, vielas e territórios florestais.

Por fim, pretende-se expandir as funcionalidades do modelo de otimização para democratizar o acesso à tecnologia e a informação, buscando baratear custos e oferecer uma concorrência balanceada no mercado para aqueles que, na fase inicial do negócio, não possuem recursos e estímulos financeiros e tecnológicos para o seu desenvolvimento.

REFERÊNCIAS

- BAKER, J. E. **Adaptive selection methods for genetic algorithms**. In: FIRST INTERNATIONAL CONFERENCE ON GENETIC ALGORITHMS AND THEIR APPLICATIONS, 1., 1985, Pittsburgh. Proceedings. Pittsburgh: Morgan Kaufmann, 1985. p. 101-111.
- BERTSIMAS, D.; TSITSIKLIS, J. Simulated Annealing. *Statist. Sci.*, v. 8, n. 1, p. 10-15, fev. 1993. Disponível em: <https://doi.org/10.1214/ss/1177011077>. Acesso em: 01 mai. 2023.
- BOURMISTROVA, A.; KHANTSIS, S. **Genetic Programming in Application to Flight Control System Design Optimisation**. In: NEW ACHIEVEMENTS IN EVOLUTIONARY COMPUTATION. InTech, 2010. DOI: 10.5772/8055.
- EIBEN, Agoston E.; SMITH, James E. **From evolutionary computation to the evolution of things**. *Nature*, v. 521, n. 7553, p. 476-482, 2015.
- EINSTEIN, A. **Zur Elektrodynamik bewegter Körper**. *Annalen der Physik*, Leipzig, v. 17, p. 891-921, 1905. Disponível em: <https://doi.org/10.1002/andp.19053221004>. Acesso em: 19 mar. 2023.
- FERREIRA JR, H. M.; TEIXEIRA, L. A. L. **A logística e a infraestrutura de transportes**. Salvador: [s.n.], 2007. 25 p.
- GAREY, M. R.; JOHNSON, D. S. **Computers and Intractability: A Guide to the Theory of NP-Completeness**. San Francisco: W. H. Freeman and Company, 1979.
- GEM. **Global Entrepreneurship Monitor: 2020/2021 Global Report**. [S.l.]: GEM Consortium, 2021. Disponível em: <https://www.gemconsortium.org/report/gem-20202021-global-report>. Acesso em: 15 mar. 2023.
- GOLDBERG, David E. **Genetic algorithms in search, optimization, and machine learning**. Addison-Wesley Longman Publishing Co., Inc., 1989.
- GOLDBERG, D. E.; DEB, K. **A comparative analysis of selection schemes used in genetic algorithms**. In: SCHAFFER, J. D. (Ed.). *Foundations of Genetic Algorithms*. San Francisco: Morgan Kaufmann Publishers, 1991. p. 69-93.
- HOLLAND, J. H. **Adaptation in natural and artificial systems**. Ann Arbor, MI: The University of Michigan Press, 1975.
- KATO, R.; PAIVA, V.; IZIDORO, S. **Algoritmos genéticos**. *Revista Bio Info*, [S.l.], ISSN 2764-8273, 9 abr. 2021. Disponível em: <https://bioinfo.com.br/algoritmos-geneticos/>. Acesso em: 23 mar. 2023.
- KATOC, S.; CHAUHAN, S. S.; KUMAR, V. **A review on genetic algorithm: past, present, and future**. *Multimedia Tools and Applications*, v. 80, p. 8091-8126, 2021.

KIM, K.-j.; HAN, I. **Genetic algorithms approach to feature discretization in artificial neural networks for the prediction of stock price index**. Expert systems with Applications, Elsevier, v. 19, n. 2, p. 125–132, 2000.

KOZA, John R. **Genetic programming: On the programming of computers by means of natural selection**. MIT Press, 1992.

LOW, C. Y.; CHONG, K. H.; SALLEH, K.; JOHNNY, K. S. P. **Path optimization using genetic algorithm evolution**. In: 2010 IEEE Student Conference on Research and Development (SCoReD). 2010.

MITCHELL, M. **An introduction to genetic algorithms**. Cambridge: MIT press, 1996.

MORAIS, G. E. S; CARVALHO, L. E. P. **Impacto Econômico da Infraestrutura de Transporte Brasileira na cadeia de suprimentos**. Revista Paramétrica, Belo Horizonte, v.14, n.2, p. 35-48, ago./dez. 2022.

MARTINS, R.; CAIXETA FILHO, J. **O desenvolvimento dos sistemas de transporte: auge, abandono e reativação recente das ferrovias**. Revista Teoria e Evidência Econômica, Passo Fundo, v. 6, n. 11, p. 137-162, 2009.

NEWTON, I. **Philosophiæ Naturalis Principia Mathematica**. Londres: Royal Society, 1687.

OBITKO, Marek. **Introduction to genetic algorithms**. Czech Technical University: Prague, Czech Republic, 1998.

ONFLEET. **Onfleet: Delivery Management Software**. Disponível em: <https://onfleet.com/>. Acesso em: 26 mar. 2023.

PACHECO, André. **Otimização por enxame de partículas – PSO**. Computação Inteligente, 2016. Disponível em: <http://computacaointeligente.com.br/algoritmos/otimizacao-por-enxame-de-particulas/>. Acesso em: 1 maio.

PACHECO, Marco A. C. **Algoritmos genéticos: princípios e aplicações**. ICA: Laboratório de Inteligência Computacional Aplicada. Departamento de Engenharia Elétrica. Pontifícia Universidade Católica do Rio de Janeiro. Fonte desconhecida, v. 28, 1999.

POTVIN, J.-Y. (1996). **Genetic algorithms for the traveling salesman problem**. Annals of Operations Research, 63(3), 337–370. doi:10.1007/bf02125403

ROUTE4ME. **Route4Me**. Disponível em: <https://www.route4me.com/>. Acesso em: 26 mar. 2023.

SEBRAE. **Quem são os pequenos negócios**. [S.l.]: SEBRAE, 2023. Disponível em: https://sebrae.com.br/sites/PortalSebrae/estudos_pesquisas/quem-sao-os-pequenos-negociosdestaque5,7f4613074c0a3410VgnVCM1000003b74010aRCRD. Acesso em: 04 abr. 202

S. BORBA, J. V.; DE OLIVEIRA GIBBON, A. R; **Modelo de custos logísticos**. SINERGIA - Revista do Instituto de Ciências Econômicas, Administrativas e Contábeis, [S. l.], v. 14, n. 2, p. 85–98, 2012. Disponível em: <https://www.periodicos.furg.br/sinergia/article/view/2131>. Acesso em: 4 abr. 2023.

SOUZA, M. J. B.; TRINDADE, F. M.; FREIRE, R. **Empreendedorismo sob o enfoque de diferentes perspectivas teóricas**. In: GIMENEZ, F. A. P.; FERREIRA, J. M.; RAMOS, S. C. (Org.). Empreendedorismo e Estratégia de Empresas de Pequeno Porte (Volume 1). Curitiba: Editora Champagnat, 2010. p. 41-54.

STOLTZ, E. **Evolution of a Salesman: A Complete Genetic Algorithm Tutorial for Python**. Towards Data Science, 2018. Disponível em: <https://towardsdatascience.com/evolution-of-a-salesman-a-complete-genetic-algorithm-tutorial-for-python-6fe5d2b3ca35>. Acesso em: 31 mar.

TRIGUEIRO, Halana Oliveira; DE FARIAS, Camilo Allyson Simões. **Otimização Estocástica Implícita e Algoritmos Genéticos como ferramenta na Operação mensal dos reservatórios Coremas - Mãe d'água**. Simpósio Brasileiro de Recursos Hídricos, v. 20, p. 1-8, 2013.

VANDERPLAS, Jake. **Python Data Science Handbook: Essential Tools for Working with Data**. O'Reilly Media, 2016.

VIEIRA, Roberta Vilhena. **Um algoritmo genético baseado em tipos abstratos de dados e sua especificação em Z**. Tese (Doutorado em Ciência da Computação) - Universidade Federal de Pernambuco, Recife, 2023. 209 p.

XIN, L., XU, P. e MANYI, G. **Logistics Distribution Route Optimization Based on Genetic Algorithm**. Comput Intell Neurosci. 2022 Jul 6;2022:8468438. doi: 10.1155/2022/8468438.

YANG, X.-S. **Genetic Algorithms**. In: NATURE-INSPIRED OPTIMIZATION ALGORITHMS, p. 77-87, 2014. DOI: 10.1016/B978-0-12-416743-8.00005-1

APÊNDICE A

Instruções de acesso e uso do modelo

Para acessar o modelo, basta apenas que o usuário acesse o seguinte link que está disponível [aqui](#). O ambiente de redirecionamento é o Google Colab, a ferramenta que o modelo foi feito. O seu uso é bem intuitivo, mas, caso o leitor não tenha familiaridade, um tutorial de uso pelo Google é disponibilizado clicando [aqui](#).

Como usar o modelo

Após acessar o link do modelo, basta clicar em "Ambiente de execução", no menu superior central e, em seguida, executar tudo (figura 20); uma alternativa a esse comando é utilizar o atalho pressionando as teclas “ctrl + F9”. Ao final da página, o leitor deve encontrar um título com o nome "Resultados". São esperados os resultados gráficos e as informações sobre a distância da rota inicial e as devidas informações de preço e otimização. Para modificar os parâmetros genéticos do modelo evolutivo, é só atribuir os valores, conforme a figura abaixo (Figura 21).

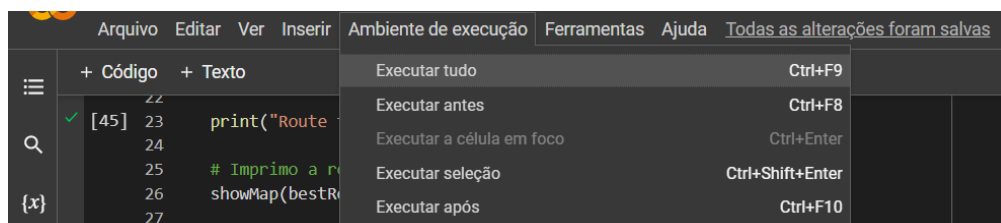


Figura 20 - Executar o modelo (Autor, 2023).

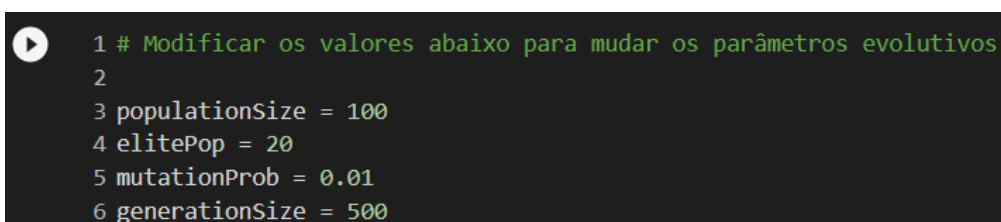


Figura 21 - Alterar os parâmetros do modelo (Autor, 2023).

Os parâmetros referem-se respectivamente, ao número de indivíduos de uma população, tamanho da população de elite, probabilidade de mutação e tamanho da geração. A mudança desses valores influencia diretamente no modelo evolutivo. Já para alterar o número de pontos, basta modificar o parâmetro “citySize” na linha 4; e para alterar a escala do gráfico, basta alterar os valores numéricos da linha 7 (figura 22).

```
1 # Modificar o número de pontos e a escala do gráfico
2
3 cityList = []
4 citySize = 25
5
6 for i in range(0, citySize):
7     cityList.append(City(x=int(random.random() * 25), y=int(random.random() * 25), z=random.random()))
```

Figura 22 - Alterar o número de pontos e a escala do gráfico (Autor, 2023).