57º SBPO
SIMPÓSIO BRASILEIRO DE PESQUISA OPERACIONAL
Gramado • RS
DE 5 A 9 DE OUTUBRO DE 2025

A EFICIÊNCIA DE UM ALGORITMO GENÉTICO PARA O PROBLEMA DE ROTEAMENTO DE VEÍCULOS CAPACITADOS

Emílio Ferreira da Silva, Maria Luiza Alves Santana, Thiago Henrique Nogueira, Gustavo Alves de Melo
Universidade Federal de Viçosa, Campus Rio Paranaíba (UFV-CRP)
Rodovia BR 230 KM 7, Rio Paranaíba - MG, 38810-000
emilio.f.silva@ufv.br, maria.santana1@ufv.br, thiagoh.nogueira@ufv.br, gustavo.a.melo@ufv.br

Maria Gabriela Mendonça Peixoto
Universidade Federal de Brasília (UnB)
Brasília, DF, 70910-900
mgabriela.unb@gmail.com

RESUMO

Este trabalho propõe a utilização de Algoritmos Genéticos (AG) para abordar o Problema de Roteamento de Veículos Capacitados (PRVC), que é um desafio clássico de otimização combinatória. O PRVC consiste em determinar as rotas mais eficientes para uma frota de veículos com capacidade limitada, de forma a minimizar a distância total percorrida e atender às demandas de um conjunto específico de clientes. Dada a complexidade do problema, que é classificado como NP-difícil, técnicas de meta-heurísticas, como o AG, são exploradas para gerar soluções de alta qualidade em um tempo computacional viável. Neste estudo, um algoritmo genético foi desenvolvido e implementado com o objetivo de avaliar sua eficácia e potencial para resolver instâncias do PRVC. A metodologia empregada envolve a geração de populações iniciais, processos de seleção, cruzamento e mutação, buscando a melhoria contínua das soluções ao longo de diversas gerações.

PALAVRAS CHAVE. Algoritmo Genético, Problema de Roteamento de Veículos Capacitados, Otimização Combinatória, Meta-heurísticas.

Tópicos: OD – Otimização Discreta, MH – Metaheurísticas.

ABSTRACT

This paper proposes the use of Genetic Algorithms (GA) to address the Capacitated Vehicle Routing Problem (CVRP), which is a classic combinatorial optimization challenge. The CVRP involves determining the most efficient routes for a fleet of vehicles with limited capacity to minimize the total distance traveled while meeting the demands of a specific set of customers. Given the complexity of the problem, which is classified as NP-hard, metaheuristic techniques such as GA are explored to generate high-quality solutions within a viable computational time. In this study, a genetic algorithm was developed and implemented with the objective of evaluating its effectiveness and potential in solving instances of the CVRP. The employed methodology involves generating initial populations, selection processes, crossover, and mutation, aiming for continuous improvement of solutions over multiple generations.

KEYWORDS. Genetic Algorithm. Capacitated Vehicle Routing Problem. Combinatorial Optimization. Metaheuristics.

Topics: OD – Discrete Optimization. MH – Metaheuristics.

DOI: 10.59254/sbpo-2025-212669
https://proceedings.science/p/212669?lang=pt-br

1. Introdução

Sendo um dos problemas mais estudados na literatura, o Problema de Roteamento de Veículos (PRV), proposto por [Dantzig e Ramser 1959] , consiste em determinar um conjunto de rotas que atendam um determinado conjunto de clientes, respeitando as restrições de demanda de cada cliente, capacidade de cada veículo e da restrição de todas as rotas se iniciarem e terminarem no depósito. O PRV é classificado como NP-difícil, sendo uma derivação do problema do Caixeiro Viajante [Baldacci and Mingozzi 2008].

Contudo, existem diversas variantes do PVR segundo as exigências operacionais da área de interesse [Laporte 1992]. O Problema de Roteamento de Veículos Capacitados (PRVC), por exemplo, é um caso da classe PVR que tem despertado grande interesse na comunidade científica nos últimos anos [Hosseinabad et al. 2017]. Ele é um problema clássico de otimização combinatória que consiste em determinar as rotas mais eficientes para uma frota de veículos com capacidade limitada além de possuir uma aplicação direta em diversas atividades econômicas, uma vez que o transporte é essencial para que os agentes econômicos realizam suas operações com o menor custo possível [Mutar et al. 2020].

Com isso, de acordo com [Bernardes 2018], a maioria dos problemas reais apresenta grafos intrincados com uma quantidade significativa de consumidores e veículos. Para encontrar soluções ótimas ou próximas das ótimas dentro de um tempo de execução razoável, são empregados métodos provenientes da otimização computacional, tais como heurísticas e meta-heurística. As meta-heurísticas avançadas permitem explorar eficientemente o espaço de soluções. O algoritmo genético proposto destaca-se pela sua capacidade de encontrar soluções de alta qualidade em um tempo computacional viável, utilizando operações genéticas como seleção, cruzamento e mutação.

Diante disso, optou-se pela utilização do Algoritmo Genético (AG) eficiente para resolver o PRVC. Essa meta-heurística evolutiva avalia múltiplos candidatos para aprimorar a qualidade da população, inspirada na evolução biológica, onde os indivíduos mais adaptados transmitem características benéficas às gerações seguintes. Um código é elaborado para gerar potenciais soluções com base em uma função objetivo, que avalia a qualidade da solução, conhecida como aptidão. O AG é então executado repetidamente até que a solução ótima seja alcançada, ou até que um número pré-determinado de gerações seja alcançado. Diante da complexidade do problema abordado, busca-se encontrar soluções para instâncias mais amplas em um tempo computacional viável.

2. Referencial Teórico

2.1. Modelo matemático para o PRVC

[Nascimento 2011] define o Problema de Roteamento de Veículos Capacitados (PRVC) como um cenário onde cada rota começa e termina em um único depósito, passando por vários clientes sem repetição de visitas. [Pereira 2012] complementa que o PRVC visa otimizar as rotas de uma frota homogênea de veículos, garantindo que a carga transportada não exceda a capacidade dos veículos, visando minimizar a distância total percorrida. O modelo matemático para resolver o PRVC, conforme descrito por [Toth e Vigo 2002], é apresentado abaixo. O modelo é representado por um grafo $G = (V, A)$, onde $V = \{0, ..., n\}$ representa os locais dos clientes, com o depósito no vértice 0, e A é o conjunto de arestas que conectam os vértices. Cada aresta (i, j), com $i \neq j$, é associada a uma matriz de valores inteiros não negativos $C = [cij]$.

Para a efetividade do modelo e para que ele possa gerar soluções ótimas, as seguintes premissas devem ser obedecidas. Cada vértice $V \setminus \{0\}$ deve ser visitado uma única vez por apenas um veículo, todas as rotas devem se iniciar e terminar no depósito, cada vértice i possui uma demanda di e a soma das demandas de qualquer uma das rotas geradas não deve exceder a capacidade Q dos veículos.

DOI: 10.59254/sbpo-2025-212669
https://proceedings.science/p/212669?lang=pt-br

Figura 1 - Modelo matemático e função objetivo

$$\text{Minimizar} \quad \sum_{i=0}^{n} \sum_{j=0}^{n} Cij \; \sum_{k=1}^{K} \mathrm{X}ijk$$

$$\text{Sujeito a} \quad \sum_{k=1}^{K} \mathrm{Y}ik = 1 \qquad (i = 1, \ldots, n)$$

$$\sum_{k=1}^{K} \mathrm{Y}ok = \mathrm{K}$$

$$\sum_{j=0}^{n} \mathrm{X}ijk = \sum_{j=0}^{n} \mathrm{X}ijk = \mathrm{Yijk} \qquad (i = 0, \ldots, n), (k = 1, \ldots, K)$$

$$\sum_{i=1}^{n} \mathrm{D}ij\mathrm{Yik} \leq |S| - 1 \qquad (\forall S \subseteq V \backslash \{0\}, |S| \geq 2, K = 1, \ldots, K)$$

$$\mathrm{X}ij \in \{0,1\} \qquad (i, j = 0, \ldots, n \,;\, i \neq j), (k = 1, \ldots, K)$$

$$\mathrm{Y}ik \in \{0,1\} \qquad (i = 0, \ldots, n), (k = 1, \ldots, K)$$

Fonte: Toth e Vigo (2002).

As restrições da função objetivo são definidas pela equação (2), onde cada vértice i deve ser visitado por um único veículo k, representado pela variável binária Yik, que é 1 quando o vértice i é visitado pelo veículo k na solução ótima, e 0 caso contrário. A equação (3) determina que os veículos disponíveis devem sair do depósito. Além disso, a equação (4) garante que um veículo que entra em um vértice i também deve sair dele. A equação (5) impõe que a capacidade Q do veículo k não seja excedida pelo somatório das demandas Di dos clientes presentes em sua rota. As equações (6) e (7) asseguram que as variáveis Xij e Yik sejam binárias. Destaca-se a importância da equação (6), que evita a formação de subcircuitos desconectados do depósito para qualquer subconjunto S de V \ {0}, limitando o número de arcos Xijk (j, i ∈ S) na solução ótima ao número de elementos do subconjunto S menos um.

2.2. Análise de trabalhos relacionados

Os Algoritmos Genéticos (AGs) representam uma meta-heurística originada do trabalho de [Holland 1975] em "Adaptation in Natural and Artificial Systems", influenciados pela teoria da evolução darwiniana. [Beasley 2002] os descreve como parte de uma categoria ampla de algoritmos de busca probabilística inteligente, inspirados no processo evolutivo observado na natureza.

Esses algoritmos operam em populações de soluções, onde cada solução é tratada como um indivíduo, comparável a um cromossomo, que é representado por um vetor de valores reais ou inteiros, dependendo do problema [Reeves 2003]. Cada indivíduo é representado por genes, determinando suas características, que são trocados ou transmitidos durante a reprodução. Segundo o mesmo autor, a seleção natural guia a evolução da população, onde os indivíduos mais aptos são selecionados para reprodução, enquanto os menos aptos são descartados.

Os AGs são amplamente aplicados no PRVC devido à sua capacidade de explorar vastos espaços de solução de forma eficaz. Eles empregam uma série de operadores, como seleção, crossover e mutação, para manipular populações de indivíduos e desenvolver soluções ao longo do tempo. Segundo [Holland 1992], os AGs são métodos de otimização adaptativa que evoluem através de gerações sucessivas, começando com uma população inicial de soluções candidatas representadas por cromossomos. No contexto do PRVC, esses cromossomos são geralmente sequências de genes que representam clientes e sua ordem de atendimento, com a função de aptidão projetada para minimizar a distância total percorrida.

2.3. Algoritmos Genéticos e sua funcionalidade no PRVC

[Lucas Caldeira et al. 2020] utilizaram a técnica de busca tabu para resolver o PRVC. Eles desenvolveram três implementações variantes do método: duas determinísticas e uma estocástica, e testaram e validaram essas implementações usando instâncias de teste da literatura.

DOI: 10.59254/sbpo-2025-212669
https://proceedings.science/p/212669?lang=pt-br

As extensões propostas e implementadas para a busca tabu alcançaram resultados ótimos para 51 das 74 instâncias testadas por eles. Mesmo nas instâncias em que não alcançaram o objetivo, os maiores erros percentuais foram de 1,40%, 1,36% e 10,31%, e o erro médio foi de 0,8%, considerando o melhor resultado obtido entre os três métodos.

Os autores [Lim et al. 2021] propuseram uma abordagem de algoritmo genético para resolver o PRV com entregas fracionarias e com múltiplos depósitos. Para o primeiro conjunto de instâncias com até 33 clientes, o AG conseguiu atingir o ótimo nas quatro instâncias. Em um segundo conjunto de instâncias, agora com dados reais, o algoritmo gerou dois resultados, o primeiro sem considerar as entregas fracionarias, gerando um resultado com 28,888km rodados e considerando o fracionamento da demanda obtiveram um resultado de 28.501km rodados.

Por fim, [Tenório 2022] propôs a abordagem de um Algoritmo Genético para solucionar o PRVC, a fim de otimizar as rotas percorridas pelos veículos de uma empresa de petrolíferos e propor uma melhor distribuição dos produtos aos clientes. Os resultados obtiveram um desempenho satisfatório em comparação com o algoritmo que já era utilizado pela empresa. Quatro exemplos foram avaliados: o primeiro apresentou uma redução de 1% na distância percorrida, o segundo 34,7%, o terceiro 31,7%, e o quarto 23,3%.

Assim, torna-se evidente a relevância da implementação de um sistema de roteamento. Essa medida não apenas aprimora a logística da empresa, tornando a entrega do produto ao cliente eficiente e eficaz, em conformidade com o pedido, mas também reduz o tempo e o custo envolvidos. Além disso, contribui significativamente para maximizar a rentabilidade global da empresa.

3. Metodologia

O PRVC busca minimizar a distância total percorrida pelos veículos, a fim de gerar uma eficiência de logística máxima. Para abordar este problema, optou-se pelo uso de Algoritmos Genéticos (AG), que são meta-heurísticas que exploram múltiplos candidatos de solução, aprimorando a qualidade da população ao longo do processo. Nesse contexto, no presente trabalho um código é desenvolvido para gerar soluções potenciais, avaliadas por uma função objetivo que determina sua qualidade, conhecida como aptidão ou fitness. O AG é iterado até alcançar uma solução ótima ou um número máximo de gerações.

Dentro do contexto dos algoritmos genéticos aplicados ao PRVC, um cromossomo é interpretado como uma solução viável para o problema. Cada solução é representada por uma matriz R. A matriz R contém informações sobre as rotas, representando os genes de cada cromossomo, juntamente com a sequência de atendimento aos clientes, logo cada linha da matriz representa uma rota, composta por uma sequência de clientes. A população do AG será composta por um vetor de soluções, formando assim, um vetor de matrizes.

Tabela 1: Representação da Solução

R: Solução:										
#1:	21	31	19	17	13	7	26			
#2:	12	01	16	30						
#3:	27	24								
#4:	29	18	08	09	22	15	10	25	05	20
#5:	14	28	11	04	23	03	02	06		

Fonte: Elaborado pelos autores (2024)

DOI: 10.59254/sbpo-2025-212669
https://proceedings.science/p/212669?lang=pt-br

A função de avaliação é responsável por calcular o Fitness de cada cromossomo e está representada pela equação abaixo. O cálculo Fitness é obtido com o somatório da variável cij que armazena a distância total percorrida pela solução.

$$min\ f(s) \sum_{i=0}^{n} \sum_{j=0}^{n} Cij$$

Diante disso, um código foi desenvolvido contendo a heurística proposta, a qual será apresentada nos próximos tópicos, e todos os algoritmos foram construídos na linguagem C++ e implementados para serem executados em ambiente Windows.

3.1. Algoritmo proposto

Segundo [Reeves e Rowe 2002], a escolha do tamanho da população precisa levar em consideração a combinação entre eficiência e eficácia. Utilizar uma população muito pequena não permite explorar com eficácia o espaço de busca, e em contrapartida uma população grande não geraria um algoritmo eficiente, uma vez que o processo computacional é mais custoso.

O Algoritmo Genético proposto para resolver o PRVC está representado no Algoritmo 1 abaixo. O algoritmo é iniciado recebendo como parâmetros o número máximo de gerações a serem geradas e o número fixo de indivíduos de uma população, definidos como 100 e 50, respectivamente. O procedimento de geração da população inicial é repetido de acordo com o parâmetro fornecido, chamando o método da seção 4.2. Após a geração da população inicial, ela é avaliada e ordenada de acordo com o melhor fitness.

O laço iniciado na linha 5 controla o número de gerações. Na linha 7, ocorre o processo de seleção dos pais (pai1 e pai2), que serão posteriormente utilizados no crossover (linha 8). A função de mutação (linha 9) realiza mudanças no gene de um cromossomo selecionado de cada indivíduo gerado pelo crossover. Após a criação da nova geração, ela é novamente avaliada e ordenada. Caso o critério de parada seja atendido, o AG é encerrado e retorna o primeiro elemento da população.

Algoritmo 1: *Ag (NumGenerations, PopulationSize)*

```
1  início
2      P ← Gera a população inicial;
3      P ← Avalia e Ordena a população por melhor valor de Fitness
4      iter ← 0
5      enquanto (iter < NumGenerations) faça
6          t ← t + 1;
7          P ← Seleciona os pais;
8          P ← Crossover();
9          P ← Mutação();
10         P ← Avalia e Ordena a população por melhor valor de Fitness;
11     fim
12     s ← Indivíduo com melhor Fitness (P);
13     Retorne s;
14 fim
```

Fonte: Elaborado pelos autores (2024)

3.2. Heurística Construtiva

A heurística construtiva proposta para gerar soluções é uma combinação de um método guloso com um fator aleatório, visando garantir diversidade na geração de soluções. O Algoritmo 2 representa essa heurística, iniciando o processo com uma lista de clientes candidatos (linha 4). O loop das linhas 5 a 10 se repetirá até que todos os clientes candidatos sejam alocados.

Inicialmente, um veículo aleatório é adicionado à primeira posição da primeira rota, seguido de um cliente aleatório na segunda posição da rota (linhas 6 e 7), conforme ilustrado na

DOI: 10.59254/sbpo-2025-212669
https://proceedings.science/p/212669?lang=pt-br

representação da solução. A função Adicionar utiliza a estratégia do vizinho mais próximo para adicionar clientes à rota até que nenhum cliente possa ser adicionado sem ultrapassar a capacidade do veículo. Posteriormente, os clientes adicionados são removidos da lista de candidatos, e o procedimento se repete até que todos sejam alocados.

Algoritmo 2: Construtivo

```
1  início
2      s a solução inicial;
3      v o número de rotas vazias;
4      LC a lista de clientes candidatados;
5      repita
6          s' ← adicionarVeiculo();
7          s' ← primeiroCliente();
8          s' ← Adicionar ();
9          s' ← atualizarLC();
10     até (LC == 0);
11     Retorne s';
12 fim
```

Fonte: Elaborado pelos autores (2024)

3.3. Crossover e Mutação

Após a geração da população inicial, o próximo passo do Algoritmo Genético (AG) é realizar crossovers entre os indivíduos, uma etapa que se baseia na adaptação do Best Cost Route Crossover (BCR) [Ombuki et al. 2006]. Inicialmente, a população é dividida em duas partes: os primeiros 25% dos indivíduos, com o melhor Fitness, são designados como "Elite", enquanto os restantes 75% são considerados "não Elite". A seleção dos pais para o crossover segue uma abordagem emparelhando o indivíduo com o melhor fitness com o de pior fitness da população não Elite. Esse emparelhamento ocorre sequencialmente, combinando o primeiro melhor com o pior, o segundo melhor com o segundo pior, e assim por diante.

Durante o processo de crossover, um par adjacente de clientes é selecionado a partir do melhor cromossomo (Pai Elite), removendo-os do segundo pai (Pai não Elite) e inserindo-os na melhor posição em relação aos demais pontos. Esse método preserva as características do Pai Elite. Essa operação é repetida para todos os 25% dos cromossomos Elite, gerando 25% de novos indivíduos que, na próxima geração, substituirão seus respectivos pais não Elite.

Ao final, a população utilizada na próxima geração consiste em: os 25% da população Elite preservada, 50% dos indivíduos que não foram pais para o crossover e 25% de novos indivíduos. O elitismo é fundamental para manter uma população controlada, com uma porção de soluções de alta qualidade. Dessa forma, ao longo das gerações, uma parcela da população total é mantida, combinada com novos indivíduos gerados pelo crossover. Essa estratégia promove a evolução contínua da população em direção a soluções ótimas.

A mutação é realizada somente nos novos indivíduos gerados pelo crossover. Para realizar a mutação dois métodos foram utilizados, o primeiro irá juntar duas rotas de um indivíduo, enquanto o segundo método irá dividir uma rota. Dessa forma, gera oportunidades de novas rotas aparecerem na população, expandindo as possíveis soluções que possam vir aparecer.

4. Resultados e discussão

Os experimentos computacionais foram conduzidos em um conjunto de instâncias propostas por [Golden et al. 1984], inicialmente destinadas ao Problema de Roteamento de Veículos Capacitados (PRVC). Os resultados desses experimentos estão documentados na Tabela 1. Das instâncias testadas, o AG conseguiu atingir o ótimo em 13 e manteve um gap médio de 1,5%. Analisando os resultados obtidos e os resultados ótimos, a diferença entre as soluções são

DOI: 10.59254/sbpo-2025-212669
https://proceedings.science/p/212669?lang=pt-br

apenas alguns clientes fora de suas rotas ótimas, mostrando que o AG consegue gerar soluções de boa qualidade. Logo novos métodos para realizar pequenas alterações na solução final gerada elo AG poderiam atingir o ótimo, como uma busca local ou até mesmo mutações.

Pela Tabela 1 é possível perceber que conforme a instância aumenta, ou seja, possui um número maior de pontos a serem alocados, o gap em relação às soluções ótimas aumenta, juntamente com o tempo de execução. Por via de testes para analisar o desempenho do AG a longo prazo dobrando os parâmetros, seu tempo computacional aumentou consideravelmente, se tornando não viável dada sua complexidade.

Tabela 1: Resultados do AG

Instância	Ótimo	Método Guloso	Busca Local	BT I		BT II		BT III		AG			Média
				custo	tempo (s)	custo	tempo (s)	custo	tempo (s)	custo	tempo (s)	gap (%)	
A-n32-k5	**784**	1308	1221	786	3.14	**784**	2.36	**784**	46.19	**784**	70,65	0,00	851
A-n33-k5	**661**	1250	1127	**661**	3.61	**661**	5.43	**661**	54.10	**661**	13,1	0,00	789
A-n33-k6	**742**	1106	1055	743	3.46	**742**	3.22	**742**	46.85	**742**	40,22	0,00	988
A-n34-k5	**778**	1041	962	787	4.93	787	3.14	**778**	56.05	**778**	68,54	0,00	1039
A-n36-k5	**799**	1109	1020	825	4.38	**799**	7.16	**799**	60.96	**799**	77,1	0,00	1086
A-n37-k5	**669**	1047	997	687	5.15	698	5.61	**669**	78.57	**669**	15,23	0,00	991
A-n37-k6	**949**	1371	1280	963	5.80	975	4.50	**949**	86.92	957	96,36	0,84	1250
A-n38-k5	**730**	1322	1232	735	5.70	734	7.94	**730**	89.04	**730**	91,42	0,00	1007
A-n39-k5	**822**	1330	1262	831	9.41	825	7.07	**822**	91.23	831	69,24	1,09	1121
A-n39-k6	**831**	1283	1189	833	6.48	834	7.70	833	90.61	835	117,61	0,48	1191
A-n44-k6	**937**	1564	1397	982	10.88	939	11.87	**937**	139.01	**937**	108,07	0,00	1214
A-n45-k6	**944**	1624	1541	969	14.78	959	21.80	948	207.11	970	52,24	2,75	1282
A-n45-k7	**1146**	1489	1465	1165	15.22	1161	11.58	**1146**	192.31	**1146**	125,3	0,00	1496
A-n46-k7	**914**	1375	1308	974	14.46	**914**	21.45	**914**	215.04	**914**	114,41	0,00	1224
A-n48-k7	**1073**	1585	1521	1117	21.44	1113	17.60	**1073**	214.60	1109	236,86	3,36	1428
A-n53-k7	**1010**	1541	1489	1054	30.94	1033	38.03	**1010**	358.20	1041	146,87	3,07	1410
A-n54-k7	**1167**	1583	1482	1189	36.02	1181	22.08	**1167**	379.14	1181	213,42	1,20	1547
A-n55-k9	**1073**	1557	1495	1111	37.78	1110	31.20	1074	488.30	**1073**	217,76	0,00	1486
A-n60-k9	**1408**	2020	1950	1367	37.81	1360	44.68	1358	512.77	**1408**	305,49	0,00	1801
A-n61-k9	**1035**	1710	1618	1036	60.09	1040	68.83	**1035**	757.95	1057	117,02	2,13	1402
A-n62-k8	**1290**	2004	1906	1320	68.21	1311	30.42	1308	602.22	1328	453,48	2,95	1758
A-n63-k9	**1634**	2312	2265	1690	45.40	1646	43.69	1627	795.14	**1634**	433,55	0,00	2065
A-n63-k10	**1315**	2178	2015	1404	46.82	1384	40.09	1325	858.95	1359	326,58	3,35	1831
A-n64-k9	**1402**	2066	2023	1432	52.45	1441	53.11	1414	855.67	1473	176,83	5,06	1941
A-n65-k9	**1177**	2030	1908	1199	85.12	**1177**	61.46	1181	821.23	1205	242,8	2,38	1619
A-n69-k9	**1168**	1771	1727	1191	68.23	1184	69.50	1163	851.63	1220	195,37	4,45	1715
A-n80-k10	**1764**	2483	2386	1827	155.09	1828	176.97	1780	1716.03	1836	397,79	4,08	2342

Fonte: Elaborado pelos autores (2024)

A análise das tabelas detalham os resultados obtidos pelo AG em diversas instâncias comparados com métodos como Guloso, Busca Local e Busca Tabu (BT), em suas variações BT I, BT II e BT III. Observou-se que em várias instâncias, o AG foi capaz de alcançar o custo ótimo, isso demonstra a eficácia do algoritmo em encontrar soluções de alta qualidade para problemas de menor complexidade.

O tempo computacional do AG aumenta significativamente com a complexidade das instâncias, especialmente quando os parâmetros são dobrados. Isso levanta questões sobre a

DOI: 10.59254/sbpo-2025-212669
https://proceedings.science/p/212669?lang=pt-br

viabilidade do AG para problemas de grande escala, onde o tempo de execução pode se tornar proibitivo. Adicionalmente, o AG geralmente se sai bem quando comparado com outras abordagens como Guloso e Busca Local, mas em algumas instâncias ele não consegue superar os métodos de Busca Tabu, especialmente em termos de custo e tempo de execução.

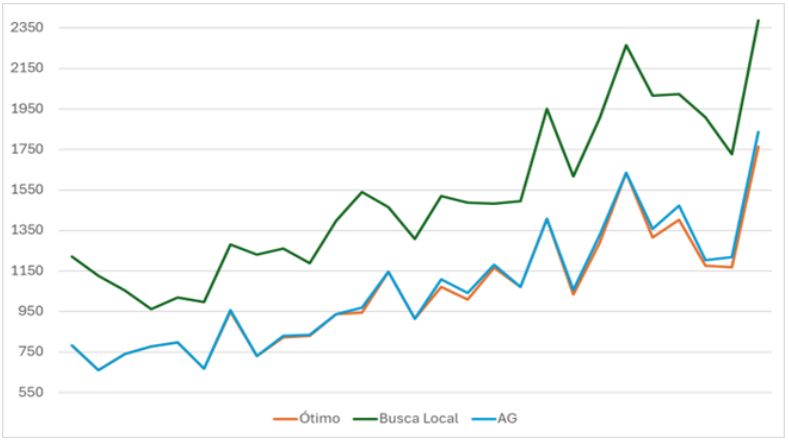
GRÁFICO 1 - Guloso x AG



Fonte: Elaborado pelos autores (2024)

Esse gráfico compara o desempenho do método Guloso com o AG, destacando a superioridade do AG em gerar soluções de alta qualidade para o PRVC. Enquanto o método Guloso oferece soluções rápidas, ele frequentemente resulta em soluções subótimas, especialmente à medida que a complexidade do problema aumenta. O AG, por outro lado, explora o espaço de soluções de maneira mais eficaz, evitando mínimos locais e alcançando resultados mais próximos do ótimo. Isso demonstra que, para problemas complexos onde a qualidade da solução é crucial, o AG é significativamente mais eficiente que o método Guloso.

GRÁFICO 2 - Busca Local x AG

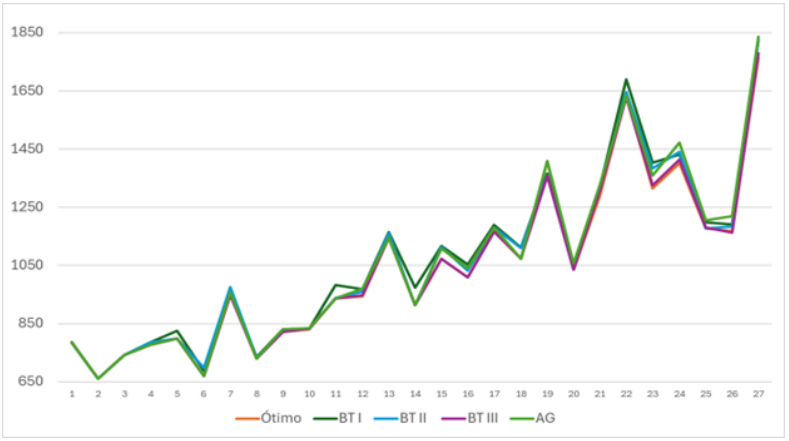


Fonte: Elaborado pelos autores (2024)

Esse gráfico compara a Busca Local e o AG, revelando que o AG supera consistentemente a Busca Local, particularmente em instâncias de maior complexidade. Embora a Busca Local ofereça melhorias em relação ao método Guloso, ela ainda fica aquém do AG na busca por soluções próximas do ótimo. Isso evidencia a superior capacidade do AG de explorar e otimizar o espaço de soluções, beneficiando-se de sua abordagem evolutiva e das operações genéticas para alcançar resultados mais precisos e eficientes.

DOI: 10.59254/sbpo-2025-212669
https://proceedings.science/p/212669?lang=pt-br

GRÁFICO 3 - BT x AG



Fonte: Elaborado pelos autores (2024)

Por fim, a comparação com as variações da Busca Tabu (BT) revela uma competição mais acirrada. Em algumas instâncias, o AG apresenta desempenho superior, mas em outras, a BT pode encontrar soluções ligeiramente melhores ou com menor tempo de execução. Este gráfico destaca que, enquanto o AG é eficaz, a BT ainda é um método muito robusto, especialmente em problemas onde a complexidade da solução cresce significativamente.

Portanto, os gráficos indicam que o Algoritmo Genético é um método eficaz para resolver o PRVC, superando métodos mais simples como o Guloso e a Busca Local em diversas instâncias. No entanto, quando comparado com métodos mais sofisticados como a Busca Tabu, o desempenho do AG é comparável, mas nem sempre superior. Essa análise gráfica sugere que o AG é particularmente vantajoso em instâncias de tamanho moderado, mas pode enfrentar desafios em termos de tempo de execução e precisão ao lidar com problemas de maior complexidade.

Esses gráficos servem para reforçar a ideia de que, embora o AG seja uma excelente escolha para o PRVC, pode haver situações em que métodos alternativos, como a Busca Tabu, ofereçam vantagens, especialmente em termos de eficiência computacional e capacidade de alcançar soluções ótimas em problemas mais desafiadores.

5. Conclusão

No decorrer deste estudo, desenvolvemos e implementamos um AG como uma meta-heurística para enfrentar o PRVC. O algoritmo foi projetado para explorar e refinar soluções através de operações genéticas, como seleção, cruzamento e mutação, melhorando continuamente a qualidade da população de soluções ao longo de várias gerações. Para avaliar a eficácia do AG, realizamos experimentos computacionais utilizando diversas instâncias do PRVC, comparando os resultados obtidos com os gerados por outros métodos tradicionais, como o Guloso, a Busca Local e a Busca Tabu.

Os resultados demonstraram que o AG tem um desempenho superior em relação ao Guloso e à Busca Local, oferecendo soluções mais próximas do custo ótimo. Essa superioridade pode ser atribuída à capacidade do AG de explorar de forma mais eficaz o espaço de soluções, o que resulta em melhores desempenhos, especialmente em problemas de tamanho moderado. No entanto, o estudo também revelou que, à medida que a complexidade das instâncias aumenta, o AG começa a apresentar um gap maior em relação às soluções ótimas, e o tempo computacional necessário para processar essas instâncias também cresce. Isso aponta para uma limitação do AG em problemas de grande escala, onde a eficiência computacional é essencial.

Ao comparar o AG com a Busca Tabu, observamos que, embora o AG seja competitivo, ele não é sempre superior. Em algumas instâncias, a Busca Tabu conseguiu resultados ligeiramente melhores ou com menor tempo de execução. Isso indica que, apesar da eficácia do

DOI: 10.59254/sbpo-2025-212669
https://proceedings.science/p/212669?lang=pt-br

AG, existe espaço para melhorias, especialmente na adaptação do algoritmo para lidar com problemas mais complexos e de grande escala.

As implicações deste trabalho sugerem que, embora o AG seja uma ferramenta poderosa para resolver o PRVC, sua aplicação em cenários de grande escala pode ser melhorada com a combinação de outras técnicas de otimização, como a Busca Tabu. Recomenda-se, para futuras pesquisas, a investigação de abordagens híbridas que possam combinar a força exploratória do AG com as vantagens de outras meta-heurísticas, além da exploração de técnicas de paralelização que possam reduzir o tempo computacional sem comprometer a qualidade das soluções.

Em conclusão, este trabalho contribui significativamente para a compreensão e aplicação dos Algoritmos Genéticos no contexto do PRVC. Embora o AG tenha demonstrado ser eficaz e robusto para resolver problemas de roteamento de veículos, os desafios permanecem em sua aplicação a instâncias maiores e mais complexas. No entanto, as descobertas aqui apresentadas fornecem uma base sólida para futuras investigações que busquem aprimorar ainda mais o uso de AGs e outras técnicas combinatórias na otimização de problemas logísticos.

Referências

Baldacci, R. and Mingozzi, A. (2008). A unified exact method for solving different classes.

Beasley, J.E., Population heuristics. In: Pardalos, P.M. & Resende, M.G.C. (Eds.), Handbook of Applied Optimization. Oxford, UK: Oxford University Press, p. 138–157, 2002.

BERNARDES, Lucas Correia et al. Metaheurísticas para o problema de roteamento de veículos capacitados: o algoritmo híbrido de otimização por colônia de formigas e a busca híbrida em grande vizinhança. 2018.

Caixeta-Filho, J. V.and Martins, R. S. (2009). Gestão logística do transporte de cargas. Atlas.

Dantzig, G. B. and Ramser, J. H. (1959). The truck dispatching problem. Management.

DE OLIVEIRA, Lucas Caldeira et al. APLICAÇÃO DA BUSCA TABU PARA SOLUÇÃO DO PROBLEMA DE ROTEAMENTO DE VEÍCULOS CAPACITADO. 2020.

FREISLEBEN, Berna; MERZ, Pedro. Novos operadores de busca genética local para o problema do caixeiro-viajante. In: Conferência Internacional sobre Solução Paralela de Problemas da Natureza. Berlim, Heidelberg: Springer Berlin Heidelberg, 1996. p. 890-899.FREISLEBEN, Berna; MERZ, Pedro. Novos operadores de busca genética local para o problema do caixeiro-viajante. In: Conferência Internacional sobre Solução Paralela de Problemas da Natureza. Berlim, Heidelberg: Springer Berlin Heidelberg, 1996. p. 890-899.

Hameed,A., Violos, J., and Leivdeas, A. (2022). A deep learning approach for iot traffic multi-classification in a smart-city scenario. IEEE Access.

HOLLAND, John H. Adaptação em sistemas naturais e artificiais: uma análise introdutória com aplicações à biologia, controle e inteligência artificial. Editora MIT, 1992.

Holland, J.H., Adaptation in natural and artificial systems. Ann Arbor, USA: University of Michigan Press, 1975.

HOSSEINABADI, Ali Asghar Rahmani; ROSTAMI, Najmeh Sadat Hosseini; KARDGAR, Maryam; MIRKAMALI,

DOI: 10.59254/sbpo-2025-212669
https://proceedings.science/p/212669?lang=pt-br

Seyedsaeid; ABRAHAM, Ajith. A new e cient approach for solving the capacitated vehicle routing problem using the gravitational emulation local search algorithm. Applied Mathematical Modelling, Elsevier, v. 49, p. 663–679, 2017.

LAPORTE, Gilbert. The vehicle routing problem: An overview of exact and approximate algorithms. European journal of operational research, Elsevier, v. 59, n. 3, p. 345–358, 1992.

LINDEN, Ricardo. Algoritmos Genéticos. 3. ed. Rio de Janeiro: Editora Ciência Moderna, 2012. 475 p.

MUTAR, M; BURHANUDDIN, M; HAMEED, A; YUSOF, N; MUTASHAR, H. An e cient improvement of ant colony system algorithm for handling capacity vehicle routing problem. International Journal of Industrial Engineering Computations, v. 11, n. 4, p. 549–564, 2020.

NASCIMENTO, I. Z. do. Abordagens determinística e estocástica para o problema de roteirização de veículos na entrega de refeições. 98 p. Dissertação (Mestrado em Ciências) - Universidade Federal do Paraná, Curitiba, 2011.

OLIVEIRA, Rodrigo Raider de. Solucionando o problema de roteamento de veículos capacitado com uso de algoritmos genéticos e paralelismo. 2023.

Ombuki, B., Ross, B. J., and Hanshar, F. (2006). Multi-objective genetic algorithms for Science, 6(1):80–91.

PEREIRA, L. G. Simulação estocástica e métodos heurísticos paralelizados para a resolução de problema de roteamento de veículos capacitados com base na estratégia “cluster first route second”. Trabalho de Conclusão de Curso (Graduação em Engenharia de Produção Civil) - Universidade Tecnológica Federal do Paraná, Curitiba, 2012.

Reeves, C.R., Genetic algorithms. In: Glover, F. & Kochenberger, G.A. (Eds.), Handbook of Metaheuristics. Boston, USA: Kluwer Academic Publishers, p. 55–82, 2003.

REEVES, C. R.; ROWE, J. E. Genetic Algorithms—Principles and Perspectives. [S.l.]: Springer US, 2002. v. 20. Https://doi.org/10.1007/b101880. ISBN 978-1-4020-7240-6.

TENÓRIO, Estela Perez da Cruz Ulhoa. Algoritmo para problema de roteamento de veículos: estudo de caso em uma empresa global líder em soluções tecnológicas para campos petrolíferos. 2022.

Toth, P. and Vigo, D. (2022). The vehicle routing problem. Monographs on Discrete Mathematics and Applications.

TOTH, P.; VIGO, D. The Vehicle Routing Problem. SIAM: Philadelphia, 2002.

DOI: 10.59254/sbpo-2025-212669
https://proceedings.science/p/212669?lang=pt-br