

# O Problema de Roteamento de Veículos Elétricos (EVRP) com Algoritmos Genéticos

Yago Pereira dos Anjos Santos

Universidade Federal de Juiz de Fora  
Programa de Pós-graduação em Modelagem Computacional

23 de julho de 2025



- 1 Introdução ao Problema
- 2 Descrição dos Métodos
- 3 Apresentação dos Resultados
- 4 Conclusão

- 1 Introdução ao Problema
- 2 Descrição dos Métodos
- 3 Apresentação dos Resultados
- 4 Conclusão

# 1. Introdução ao Problema

- **Problema de Roteamento de Veículos (VRP):**

- Definir as rotas ideais para uma frota de veículos visitar um conjunto de clientes e retornar a um depósito central.
- Objetivo: Minimizar a distância total percorrida.

- **Relevância:** Logística, transporte, entrega, coleta de lixo, etc.

- **Tipo de Problema:** NP-Hard.

- **Abordagem:** Utilizar Algoritmos Genéticos (AGs) para encontrar soluções aproximadas.

- **Foco atual:** Uma variação simplificada, onde **um único veículo** deve visitar todos os clientes (Problema do Caixeiro Viajante - TSP), sem considerar restrições de capacidade ou bateria.

- 1 Introdução ao Problema
- 2 Descrição dos Métodos
- 3 Apresentação dos Resultados
- 4 Conclusão

## 2. Descrição dos Métodos: Algoritmo Genético

- **Algoritmos Genéticos (AGs):**

- Meta-heurísticas inspiradas na evolução biológica.
- Buscam otimizar problemas complexos através de população, seleção, cruzamento e mutação.

- **Representação do Indivíduo:**

- **Permutação Direta dos Clientes:** Cada indivíduo é uma lista com os IDs dos clientes na ordem em que serão visitados.
- Exemplo: '[3, 1, 5, 2, 4]' (visitar cliente 3, depois 1, etc.)
- Vantagem: Garante soluções válidas (todos os clientes são visitados uma única vez).

## 2. Descrição dos Métodos: Operadores Genéticos

### ● Função de Aptidão (Fitness):

- Inverso da distância total da rota percorrida pelo(s) veículo(s).
- $Fitness = 1/Distância\_Total$  (quanto menor a distância, maior a aptidão).
- Atualmente: Soma da distância percorrida pelo **único veículo** saindo do depósito, passando por todos os clientes na ordem da permutação, e retornando ao depósito.

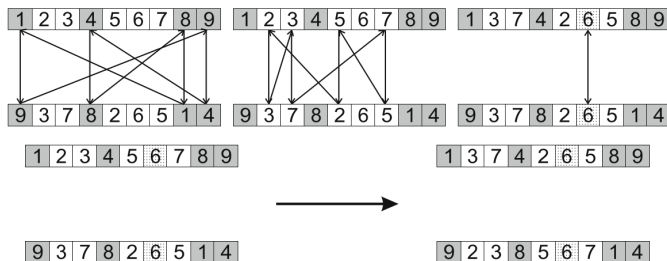
### ● Seleção de Pais: Torneio (Tournament Selection)

- $k$  indivíduos são selecionados aleatoriamente da população.
- O indivíduo com maior aptidão entre os  $k$  é escolhido como pai.
- Processo repetido para escolher o segundo pai.
- Vantagem: Controla a pressão seletiva (maior  $k$ , maior pressão).

## 2. Descrição dos Métodos: Operadores Genéticos (Cont.)

### ● Recombinação por Ciclos (Cycle Crossover)

- Ideal para permutações, preserva a posição relativa dos elementos.
- Identifica ciclos de elementos entre os dois pais e os troca para gerar os filhos.
- Garante que os filhos também sejam permutações válidas.



**Figura 1:** Recombinação por ciclos: identificação de ciclos e geração de novos indivíduos. Fonte: Adaptado de [1].



## 2. Descrição dos Métodos: Operadores Genéticos (Cont.)

### • **Mutação por Inversão (Inversion Mutation)**

- Seleciona aleatoriamente dois pontos na permutação.
- Inverte a ordem da sub-sequência entre esses pontos.
- Ajuda a explorar o espaço de busca evitando mínimos locais.



Figura 2: Representação de uma mutação por inversão. Fonte: Adaptado de [1].

- **Elitismo:** O melhor indivíduo da geração atual é sempre copiado para a próxima população.

- 1 Introdução ao Problema
- 2 Descrição dos Métodos
- 3 Apresentação dos Resultados**
- 4 Conclusão

### 3. Apresentação dos Resultados

- **Instância Testada:** 'E-n23-k3.evrp' (Tratada como TSP para 1 veículo).
- **Parâmetros do AG:**
  - População: 100 indivíduos
  - Taxa de Mutação: 2%
  - Máximo de Avaliações por execução:  $25000 \times 23 = 575000$
  - Tamanho do Torneio ( $k$ ): 9
  - Número de Execuções: 20

instância	Resultados AG			
	min	max	mean	stdev
E-n23-k3.evrp	470.06	554.97	485.96	18.30

Tabela 1: Resultados obtidos a partir da implementação do AG

### 3. Apresentação dos Resultados

- **Instância Testada:** 'E-n51-k5.evrp' (Tratada como TSP para 1 veículo).
- **Parâmetros do AG:**
  - População: 100 indivíduos
  - Taxa de Mutação: 2%
  - Máximo de Avaliações por execução:  $25000 \times 51 = 1275000$
  - Tamanho do Torneio ( $k$ ): 9
  - Número de Execuções: 20

instância	Resultados AG			
	min	max	mean	stdev
E-n51-k5.evrp	444.58	482.98	461.35	10.58

Tabela 2: Resultados obtidos a partir da implementação do AG

### 3. Apresentação dos Resultados (Cont.)

- No trabalho anterior:

instâncias	Resultados Anteriores AGB			
	min	max	mean	stdev
E-n23-k3.evrp	918.56	918.56	918.56	0.0
E-n51-k5.evrp	1313.47	1313.47	1313.47	0.0

Tabela 3: Resultados obtidos anteriormente pelo AGB

- Instância E-n23-k3.evrp:
  - Diferença percentual (ref. ao valor ótimo 470.06):  $-48.83\%$ .
- Instância E-n51-k5.evrp:
  - Diferença percentual (ref. ao valor ótimo 444.58):  $-66.15\%$ .

- 1 Introdução ao Problema
- 2 Descrição dos Métodos
- 3 Apresentação dos Resultados
- 4 Conclusão**

## 4. Conclusão

- A escolha da representação é crucial para a eficácia de um AG em problemas combinatórios como o EVRP.
- Com o problema simplificado (TSP com 1 veículo, sem restrições), o algoritmo demonstra a capacidade de encontrar boas soluções em um número limitado de avaliações.
- Importante atentar-se para os tipos de operadores genéticos adequados.



Eiben, A. E., & Smith, J. E. (2015). *Introduction to evolutionary computing*. Springer.



Mavrovouniotis, M., & Menelaou, M. (2020). EVRP Competition 2020. Disponível em:  
<https://mavrovouniotis.github.io/EVRPcompetition2020/>  
[Acessado em 23 de julho de 2025].



Obrigado!