

# Implementação e Análise de Algoritmos Genéticos para o Problema da Mochila: Comparação de Operadores Genéticos

Klaus Siegfried Beckmann<sup>1</sup>

<sup>1</sup>Universidade Tuiuti do Paraná  
Curitiba – Paraná

klaus.beckmann@utp.edu.br

**Resumo.** *Este trabalho apresenta a implementação e análise de Algoritmos Genéticos (AGs) para resolver o Problema da Mochila (Knapsack Problem). Foram testadas quatro configurações diferentes dos operadores genéticos: crossover (um ponto, dois pontos, uniforme), taxas de mutação (baixa, média, alta), inicialização da população (aleatória vs. heurística) e critérios de parada (gerações fixas vs. convergência). Os experimentos foram conduzidos em 10 instâncias do problema, com múltiplas execuções para garantir significância estatística. Os resultados demonstram que a configuração com crossover de dois pontos e inicialização heurística apresenta o melhor desempenho geral, enquanto o critério de convergência reduz significativamente o tempo de execução sem comprometer a qualidade das soluções.*

## 1. Introdução

O Problema da Mochila (Knapsack Problem) é um problema clássico de otimização combinatória que consiste em selecionar um subconjunto de itens, cada um com peso e valor específicos, de forma a maximizar o valor total sem exceder a capacidade de peso da mochila. Este problema é NP-difícil e possui aplicações práticas em diversas áreas, como logística, alocação de recursos e otimização de investimentos.

Algoritmos Genéticos (AGs) são meta-heurísticas inspiradas no processo de evolução natural, utilizando conceitos como seleção, crossover e mutação para evoluir uma população de soluções candidatas em direção a soluções ótimas ou próximas do ótimo. A escolha de AGs para resolver o Problema da Mochila é justificada pela capacidade desses algoritmos de lidar eficientemente com espaços de busca discretos e sua robustez em evitar mínimos locais.

O objetivo deste trabalho é implementar um Algoritmo Genético para o Problema da Mochila e analisar comparativamente o impacto de diferentes configurações dos operadores genéticos no desempenho do algoritmo, considerando tanto a qualidade das soluções encontradas quanto o tempo de execução.

## 2. Metodologia

### 2.1. Representação do Problema

O Problema da Mochila foi modelado utilizando uma representação binária, onde cada gene do cromossomo indica se um item específico está incluído (1) ou não (0) na solução.

Para uma instância com  $n$  itens, cada indivíduo é representado por um vetor binário de tamanho  $n$ .

A função objetivo a ser maximizada é dada por:

$$f(x) = \sum_{i=1}^n v_i \cdot x_i \quad (1)$$

sujeita à restrição de capacidade:

$$\sum_{i=1}^n w_i \cdot x_i \leq W \quad (2)$$

onde  $v_i$  é o valor do item  $i$ ,  $w_i$  é o peso do item  $i$ ,  $x_i \in \{0, 1\}$  indica a inclusão do item, e  $W$  é a capacidade da mochila.

## 2.2. Implementação do Algoritmo Genético

O AG implementado possui os seguintes componentes principais:

**Função de Fitness:** Utiliza penalização para soluções inviáveis, aplicando uma penalidade proporcional ao excesso de peso quando a capacidade é ultrapassada.

**Seleção:** Implementada através de torneio com tamanho 3, proporcionando um bom equilíbrio entre pressão seletiva e diversidade populacional.

**Elitismo:** Mantém os 10% melhores indivíduos da população a cada geração, garantindo que as melhores soluções não sejam perdidas.

## 2.3. Configurações Testadas

Foram implementadas e testadas quatro configurações distintas do AG:

### Configuração 1 - Base:

- Crossover: Um ponto
- Taxa de mutação: 0.01 (baixa)
- Inicialização: Aleatória
- Critério de parada: 200 gerações

### Configuração 2 - Dois Pontos:

- Crossover: Dois pontos
- Taxa de mutação: 0.05 (média)
- Inicialização: Aleatória
- Critério de parada: 200 gerações

### Configuração 3 - Uniforme:

- Crossover: Uniforme
- Taxa de mutação: 0.1 (alta)
- Inicialização: Aleatória
- Critério de parada: 200 gerações

### Configuração 4 - Heurística:

- Crossover: Um ponto
- Taxa de mutação: 0.02 (baixa)
- Inicialização: Heurística (baseada na razão valor/peso)
- Critério de parada: Convergência (30 gerações sem melhoria)

### 2.4. Experimentos

Os experimentos foram conduzidos utilizando 10 instâncias do Problema da Mochila com diferentes características. Para cada configuração e instância, foram realizadas 5 execuções independentes, totalizando 200 execuções. Os parâmetros comuns incluem população de 100 indivíduos e taxa de crossover de 0.8.

## 3. Resultados

### 3.1. Análise Geral dos Resultados

A Tabela 1 apresenta um resumo estatístico dos resultados obtidos pelas quatro configurações testadas.

**Tabela 1. Resumo estatístico dos resultados por configuração**

Configuração	Fitness Médio	Desvio Padrão	Tempo Médio (s)
Base - Um Ponto + Mutação Baixa	2017.70	945.9	0.54
Dois Pontos + Mutação Média	2025.10	945.9	0.55
Uniforme + Mutação Alta	2024.40	945.9	0.45
Heurística + Convergência	2025.10	945.9	0.07

### 3.2. Resultados Detalhados por Instância

A Tabela 2 mostra os resultados detalhados para algumas instâncias representativas do conjunto de testes.

**Tabela 2. Resultados detalhados por instância selecionada**

Instância	Base	Dois Pontos	Uniforme	Heurística
knapsack_1	973	973	973	973
knapsack_5	1468.40	1491	1491	1491
knapsack_8	2921.80	2957	2957	2957
knapsack_10	3456.80	3473	3466	3473

### 3.3. Análise de Convergência

A configuração com critério de convergência apresentou um comportamento interessante, convergindo consistentemente entre as gerações 29-55, com média de 32 gerações. Isso representa uma redução significativa no tempo de execução (87% menor) comparado às outras configurações, mantendo a mesma qualidade de soluções.

## 4. Discussão

### 4.1. Impacto dos Operadores de Crossover

Os resultados mostram que o crossover de dois pontos apresentou melhor desempenho que o crossover de um ponto, obtendo consistentemente os valores ótimos em todas as instâncias. O crossover uniforme mostrou desempenho intermediário, sugerindo que para este problema específico, a preservação de blocos contíguos de genes (característica do crossover de dois pontos) é benéfica.

## **4.2. Efeito das Taxas de Mutação**

Contrariamente ao esperado, taxas de mutação mais altas não necessariamente prejudicaram o desempenho. A configuração com mutação alta (0.1) apresentou resultados competitivos, indicando que para o Problema da Mochila, uma exploração mais agressiva do espaço de busca pode ser vantajosa, especialmente considerando a natureza discreta do problema.

## **4.3. Inicialização Heurística vs. Aleatória**

A inicialização baseada na heurística de razão valor/peso demonstrou ser extremamente eficaz, não apenas mantendo a qualidade das soluções, mas reduzindo drasticamente o tempo de execução através do critério de convergência. Isso sugere que incorporar conhecimento específico do domínio na inicialização pode acelerar significativamente a convergência do AG.

## **4.4. Critérios de Parada**

O critério de convergência mostrou-se superior ao critério de gerações fixas, reduzindo o tempo de execução em 87% sem comprometer a qualidade das soluções. Isso indica que o algoritmo consegue identificar quando atingiu um ótimo local/global e pode parar prematuramente, economizando recursos computacionais.

## **4.5. Análise de Escalabilidade**

Observou-se que para instâncias menores (como knapsack\_1 com apenas 5 itens), todas as configurações convergiram para a solução ótima. Para instâncias maiores, as diferenças entre configurações tornaram-se mais evidentes, demonstrando a importância da escolha adequada dos operadores genéticos em problemas de maior complexidade.

## **5. Conclusão**

Este estudo comparativo demonstrou que a escolha dos operadores genéticos tem impacto significativo no desempenho de Algoritmos Genéticos aplicados ao Problema da Mochila. As principais conclusões são:

1. A configuração "Dois Pontos + Mutação Média" apresentou o melhor desempenho em termos de qualidade de solução, obtendo consistentemente os valores ótimos.
2. A configuração "Heurística + Convergência" ofereceu o melhor custo-benefício, mantendo alta qualidade das soluções com redução drástica no tempo de execução (87% menor).
3. A inicialização heurística baseada na razão valor/peso mostrou-se extremamente eficaz para acelerar a convergência do algoritmo.
4. O critério de parada por convergência é superior ao critério de gerações fixas, otimizando o uso de recursos computacionais.

### **5.1. Sugestões para Trabalhos Futuros**

Como trabalhos futuros, sugere-se:

- Implementação de operadores de crossover híbridos que combinem as vantagens observadas;

- Teste com populações de diferentes tamanhos para avaliar o trade-off entre diversidade e tempo de execução;
- Aplicação de técnicas de auto-adaptação de parâmetros durante a execução do AG;
- Comparação com outras meta-heurísticas como Algoritmos de Enxame de Partículas e Algoritmos de Colônia de Formigas;
- Teste em instâncias de maior escala para validar a escalabilidade das configurações propostas.

O estudo demonstrou que AGs são eficazes para resolver o Problema da Mochila, e que a configuração adequada dos operadores genéticos pode resultar em melhorias significativas tanto na qualidade das soluções quanto na eficiência computacional.

Artigo elaborado com o auxílio de IA para correção ortográfica.