

# Revenimento Paralelo Aplicado ao Problema de Minimização de Pilhas Abertas

Mauro Lúcio Afonso Paulino dos Santos Filho

Departamento de Computação  
Instituto de Ciências Exatas e Biológicas  
Universidade Federal de Ouro Preto

1 de setembro de 2025



- 1 Introdução
- 2 Problema
- 3 Método
- 4 Desenvolvimento
- 5 Experimentos e Resultados
- 6 Conclusão e Trabalhos Futuros

# Contexto

- ▶ Ambiente industrial;
- ▶ Armazenagem e manuseio;
- ▶ Otimização.



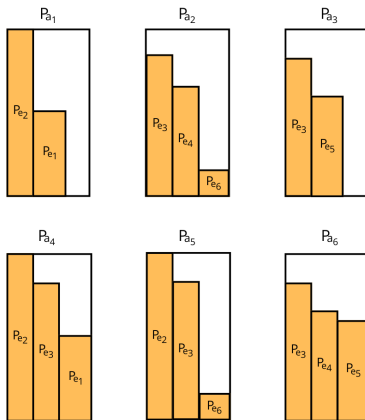
# Definições

- ▶ Padrão: esquema de corte;
- ▶ Peça: unidade produzida a partir do corte;
- ▶ Pilhas: agrupamento temporário de peças.



## Objetivo

Sequenciar os padrões de corte de forma a minimizar o número máximo de pilhas abertas simultaneamente.



**Tabela:** Matriz de incidência ( $A$ ).

Peça/Padrão	$P_{a_1}$	$P_{a_2}$	$P_{a_3}$	$P_{a_4}$	$P_{a_5}$	$P_{a_6}$
$P_{e_1}$	1	0	0	1	0	0
$P_{e_2}$	1	0	0	1	1	0
$P_{e_3}$	0	1	1	1	1	1
$P_{e_4}$	0	1	0	0	0	1
$P_{e_5}$	0	0	1	0	0	1
$P_{e_6}$	0	1	0	0	1	0

# Matriz de Pilhas Abertas 1

## Exemplo de solução

$$\pi_1 = [P_{a_1}, P_{a_2}, P_{a_3}, P_{a_4}, P_{a_5}, P_{a_6}]$$

**Tabela:** Matriz de pilhas abertas ( $B^{\pi_1}$ ).

Pilha/Estágio	1	2	3	4	5	6
$P_{e_1}$	1	1	1	1	0	0
$P_{e_2}$	1	1	1	1	1	0
$P_{e_3}$	0	1	1	1	1	1
$P_{e_4}$	0	1	1	1	1	1
$P_{e_5}$	0	0	1	1	1	1
$P_{e_6}$	0	1	1	1	1	0

# Matriz de Pilhas Abertas 2

## Exemplo de solução

$$\pi_2 = [P_{a_3}, P_{a_6}, P_{a_2}, P_{a_5}, P_{a_4}, P_{a_1}]$$

**Tabela:** Matriz de pilhas abertas ( $B^{\pi_2}$ ).

Pilha/Estágio	1	2	3	4	5	6
$P_{e_1}$	0	0	0	0	1	1
$P_{e_2}$	0	0	0	1	1	1
$P_{e_3}$	1	1	1	1	1	0
$P_{e_4}$	0	1	1	0	0	0
$P_{e_5}$	1	1	0	0	0	0
$P_{e_6}$	0	0	1	1	0	0



## Artigo

Frinhani, Carvalho e Soma (2018)

## Principais Contribuições

- ▶ PageRank e grafo MOSP;
- ▶ Instâncias significativamente maiores (cerca de 5 vezes maiores que as anteriores), ampliando o escopo dos testes;
- ▶ O método superou heurísticas clássicas em termos de qualidade das soluções e tempo de execução.

## Fundamentação Teórica

- ▶ MCMC;
- ▶ Algoritmo de Metropolis;
- ▶ Distribuição de Boltzmann;
- ▶ Algoritmo de Metropolis-Hastings.

## Justificativa

O PT permite explorar amplamente o espaço de soluções;

Sua implementação paralela possui bons resultados preliminares, mostrando que o método alcança soluções competitivas.

# Propostas de Troca de Temperatura

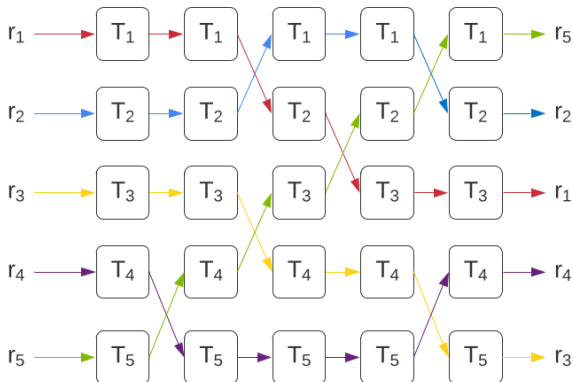
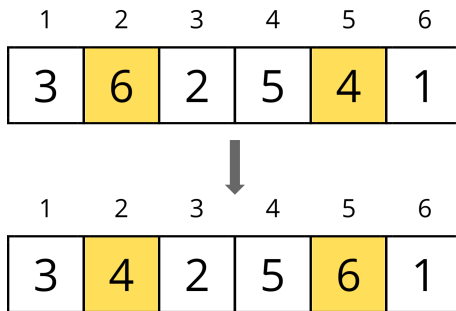


Figura: Almeida, de Castro Lima e Carvalho (2025)

# Soluções Iniciais e Codificação/Decodificação

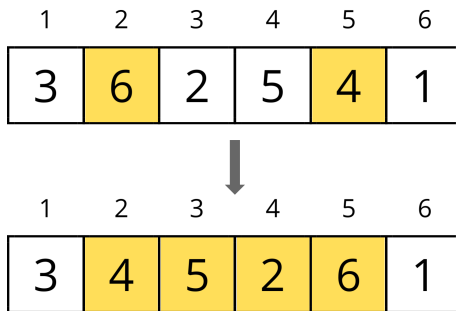
- ▶ Solução aleatória;
  - ▶ MCNH.
- 
- ▶ Sequenciamento de Padrões;
  - ▶ Sequenciamento de Peças.

# Estruturas de Vizinhança Swap



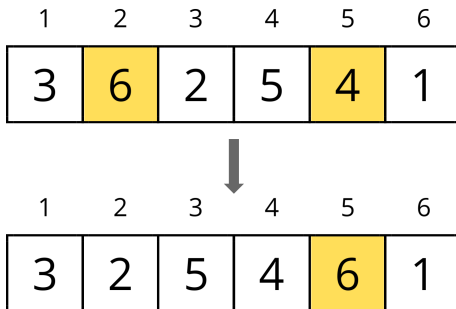
**Figura:** Exemplo de movimento *swap* entre as posições 2 e 5.

# Estruturas de Vizinhança 2-opt



**Figura:** Exemplo de movimento 2-opt entre as posições 2 e 5.

# Estruturas de Vizinhança Inserção aleatória



**Figura:** Exemplo de movimento de inserção do elemento da posição 2 na posição 5.

## Configurações

- ▶ Intel i7-10700 (8 núcleos / 16 threads), 16 GB RAM, Ubuntu 20.04;
- ▶ Implementação em C++ com compilador g++ -O3;
- ▶ PT configurado para 5 threads;
- ▶ Execuções: 5 repetições para cada instância.

## Instâncias

- ▶ 610 Instâncias;
- ▶ 400x400; 600x600; 800x800; 1000x1000;
- ▶ Conjuntos 2, 4, 6, 8, 10, 14, 18, 20, 24, 28, 30, 34, 38, 40, 44, 48, 50, 54.



Parâmetro	Valores possíveis
$T_0$ (temperatura inicial)	{ <b>0,01</b> ; 0, 1; 1}
$T_f$ (temperatura final)	{3; 5; <b>10</b> }
$N_{kmax}$ (comprimento da cadeia de Markov)	{300; 400; 500; <b>600</b> }
Distribuição de temperatura <sup>a</sup>	{1; <b>2</b> ; 3; 4}
Forma de gerar soluções vizinhas <sup>b</sup>	{0; 1; <b>2</b> }
Tipo de atualização de temperatura <sup>c</sup>	{ <b>0</b> ; 1; 2; 3}
Taxa de ajuste das temperaturas <sup>d</sup>	{3; 4; <b>5</b> }

<sup>a</sup> 1: linear, 2: linear inteira, 3: exponencial, 4: geométrica.

<sup>b</sup> 0: swap, 1: 2-opt, 2: inserção aleatória.

<sup>c</sup> 0: sem ajuste; 1: ajusta para taxa de troca  $\approx 23,4\%$ , 2: equaliza probabilidade de troca entre réplicas vizinhas, 3: otimiza o tempo de ida e volta entre temperaturas extremas (ALMEIDA; de Castro Lima; CARVALHO, 2025).

<sup>d</sup> Calculada como o número total de trocas dividido pelo valor da taxa.

## Comparação com Estado da Arte

- ▶ Melhor desempenho do PT-MOSP em praticamente todos os casos (2,4% de gap médio);
- ▶ Ganhos mais expressivos em instâncias menos densas (até 14 peças/padrão, gap de até 15,55%, com 5% de gap médio);
- ▶ Tempo de execução: de segundos até 15 minutos.

## Estatísticas

- ▶ Shapiro-Wilk indica não-normalidade;
- ▶ Wilcoxon  $p < 0.001$  para todos os tamanhos;
- ▶ Indica diferença estatística entre os métodos.

# Conclusão


## Conclusão


- ▶ PT-MOSP supera consistentemente o PieceRank;
- ▶ Vantagem mais relevante em instâncias menos densas.

## Trabalhos Futuros

- ▶ Melhorias na implementação paralela;
- ▶ Novas estratégias de vizinhança;
- ▶ Aplicação a problemas combinatórios correlatos.



 ALMEIDA, A. L. B.; de Castro Lima, J.; CARVALHO, M. A. M. Revisiting the parallel tempering algorithm: High-performance computing and applications in operations research. *Computers & Operations Research*, v. 178, p. 107000, 2025. ISSN 0305-0548. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0305054825000280>>.

 FRINHANI, R. d. M. D.; CARVALHO, M. A. M.; SOMA, N. Y. A pagerank-based heuristic for the minimization of open stacks problem. *Plos one*, Public Library of Science San Francisco, CA USA, v. 13, n. 8, p. e0203076, 2018.