

Estratégias Avançadas de Simulated Annealing para o Problema do Caixeiro Viajante

Letícia Capitani Trapp Sampaio¹

¹Departamento de Ciência da Computação

Universidade do Estado de Santa Catarina (UDESC) - Joinville, SC - Brazil

leticia.sampaio21@edu.udesc.br

Resumo. Este trabalho investiga configurações do Simulated Annealing para o Problema do Caixeiro Viajante, comparando operadores de vizinhança, estratégias de resfriamento e reaquecimento adaptativo em instâncias TSPLIB. Os experimentos identificam que o parâmetro SA_{max} é sensível ao tamanho da instância, o reaquecimento beneficia instâncias maiores com schedules não lineares, e o operador 2-opt híbrido supera significativamente o swap simples.

1. Introdução

O Problema do Caixeiro Viajante é um problema de otimização combinatória NP-difícil que consiste em encontrar a rota de menor custo visitando todas as cidades exatamente uma vez e retornando à origem. Sua complexidade exponencial torna métodos exatos impraticáveis para instâncias grandes, motivando o uso de meta-heurísticas.

O Simulated Annealing (SA), proposto por Kirkpatrick et al. (1983) [2], é uma meta-heurística eficaz para o TSP, inspirada no recocimento de metais. Sua principal vantagem é a capacidade de escapar de mínimos locais através do critério de aceitação probabilístico [1], permitindo exploração efetiva do espaço de busca.

Este trabalho investiga sistematicamente diferentes configurações do SA, comparando operadores de vizinhança, estratégias de resfriamento e mecanismos de reaquecimento adaptativo, buscando identificar as configurações mais eficazes para instâncias de diferentes tamanhos.

2. Metodologia de Desenvolvimento

A metodologia empregada partiu da implementação clássica do SA, sendo progressivamente refinada através de experimentos sistemáticos. Foram utilizadas duas instâncias do formato TSPLIB contendo coordenadas cartesianas bidimensionais: eil51 (51 cidades) e eil101 (100 cidades). Os custos ótimos conhecidos para estas instâncias são 426 e 21282, respectivamente, servindo como referência para avaliar a qualidade das soluções.

2.1. Implementação do Simulated Annealing

2.1.1. Carregamento e Representação das Instâncias

Após o carregamento das coordenadas, foi construída uma matriz de distâncias $D \in \mathbb{R}^{n \times n}$, onde $D[i][j]$ representa a distância euclidiana entre as cidades i e j . Uma rota é representada como uma permutação $\pi = [\pi_0, \pi_1, \dots, \pi_{n-1}]$ dos índices das cidades, onde π_0 é fixado como ponto de partida e retorno. O custo de uma rota corresponde ao somatório das distâncias entre cidades consecutivas na permutação, incluindo o retorno de π_{n-1} a π_0 .

2.1.2. Operadores de Vizinhança

Foram explorados e testados dois operadores de geração de vizinhança para fins de comparação (Seção 4.3):

Operador Swap. A primeira função de geração de vizinhança utiliza o operador de *swap* simples, trocando aleatoriamente duas cidades da rota (exceto a primeira, que permanece fixa como ponto de partida). Esta escolha proporciona um equilíbrio entre simplicidade de implementação e eficácia na exploração do espaço de soluções. O operador pode ser descrito como:

$$\text{swap}(s, i, j) : \quad s' = s \text{ com } s'[i] \leftrightarrow s'[j], \quad i, j \in \{1, 2, \dots, n - 1\}, \quad i \neq j \quad (1)$$

Operador 2-opt. O segundo operador remove dois arcos da rota e reconecta os segmentos invertendo um deles, eliminando potenciais cruzamentos na rota. Formalmente, o operador inverte um segmento da rota entre as posições i e j :

$$2\text{-opt}(s, i, j) : \quad s' = [s[0], \dots, s[i-1], s[j], s[j-1], \dots, s[i], s[j+1], \dots, s[n-1]] \quad (2)$$

Estratégia Híbrida. Na implementação final, ambos os operadores foram combinados, ou seja, em cada iteração, o operador 2-opt é aplicado com probabilidade de 70%, enquanto o *swap* é aplicado nos 30% restantes. Esta estratégia busca aproveitar a capacidade do 2-opt de realizar mudanças mais significativas na estrutura da rota, ao mesmo tempo que mantém a diversificação proporcionada pelo operador mais simples. A escolha probabilística entre os operadores permite que o algoritmo alterne entre movimentos de maior e menor impacto, equilibrando intensificação e diversificação da busca.

2.1.3. Critério de Aceitação de Metropolis

No SA, é definido nativamente que uma nova solução é sempre aceita se melhora o custo atual ($\Delta < 0$). Para soluções piores ($\Delta \geq 0$), a probabilidade de aceitação é dada por:

$$P(\text{aceitar}) = e^{-\Delta/T} \quad (3)$$

onde $\Delta = f(s') - f(s)$ é a diferença de custo e T é a temperatura corrente.

2.1.4. Parâmetro SA_{max}

Foi implementado um parâmetro dinâmico SA_{\max} que define quantas iterações são executadas em cada nível de temperatura antes do resfriamento.

Para investigar o impacto deste parâmetro na qualidade das soluções, foram realizados experimentos sistemáticos (Seção 4.1) com diferentes valores de SA_{\max} :

- Para a instância de 51 cidades, foram testados os valores $SA_{\max} \in \{3, 7\}$
- Para a instância de 100 cidades, foram testados os valores $SA_{\max} \in \{7, 15\}$

2.2. Estratégias de Resfriamento (*Cooling Schedules*)

Cinco diferentes estratégias de resfriamento foram implementadas e comparadas. Cada *schedule* define como a temperatura $T(i)$ evolui ao longo das iterações i , onde N é o número máximo de iterações, T_0 é a temperatura inicial e T_{\min} é a temperatura mínima. A escolha do *cooling schedule* é crucial para o desempenho do SA, pois controla o equilíbrio entre exploração e *exploitation*.

Schedule 0 (Linear): $T(i) = T_0 - i \times \frac{T_0 - T_{\min}}{N}$

Schedule 5 (Cosseno): $T(i) = \frac{1}{2}(T_0 - T_{\min}) \times (1 + \cos(\frac{i\pi}{N})) + T_{\min}$

Schedule 6 (Tangente Hiperbólica): $T(i) = \frac{1}{2}(T_0 - T_{\min}) \times (1 - \tanh(\frac{10i}{N} - 5)) + T_{\min}$

Schedule 8 (Exponencial): $T(i) = T_0 \times e^{-\frac{1}{N} \ln(\frac{T_0}{T_{\min}}) \times i}$

Schedule 9 (Exponencial Quadrático): $T(i) = T_0 \times e^{-\frac{1}{N^2} \ln(\frac{T_0}{T_{\min}}) \times i^2}$

2.3. Mecanismo de Reaquecimento Progressivo

Para evitar convergência prematura, foi implementado um mecanismo de reaquecimento adaptativo. Quando o contador de iterações sem melhoria atinge $L_{\text{stag}} = 80000$, a temperatura é elevada para:

$$T_{\text{reheat}} = 0.3 \times T_0 \quad (4)$$

A partir deste ponto, aplica-se resfriamento geométrico:

$$T_{\text{reheat}}(k) = T_{\text{reheat}} \times \alpha^k \quad (5)$$

onde $\alpha = 0.95$ para 51 cidades e $\alpha = 0.98$ para 100 cidades. O reaquecimento cessa quando $T_{\text{reheat}}(k) \leq T_{\text{base}}$, retornando ao *schedule* original. A diferenciação das taxas reflete o tamanho do espaço de busca: espaços menores requerem resfriamento mais rápido, enquanto espaços maiores necessitam de descida gradual para exploração efetiva.

3. Experimentos e Parâmetros

Foram realizados cinco experimentos para avaliar progressivamente diferentes configurações:

Tabela 1. Configuração dos experimentos para ambas as instâncias

Nº	Operador	Reaquec.	SA _{max}		Seeds	Objetivo
			51 cid.	100 cid.		
1	Swap	Não	7	15	10	Baseline
2	Swap	Não	3	7	1	Análise SA _{max}
3	Swap	Não	7	15	1	Análise SA _{max}
4	Swap	Sim	7	15	1	Teste reaquec.
5	2-opt	Sim	7	15	10	Config. final

Para os experimentos 1 e 5, foram utilizadas 10 sementes aleatórias {42, 123, 456, 789, 1011, 1314, 1617, 1920, 2223, 2526} para análise estatística, os demais utilizaram apenas a seed 1617. Isso permitiu extrair média, mediana e desvio padrão por *schedule*.

3.1. Configuração Experimental

Os parâmetros foram calibrados de acordo com o tamanho da instância, conforme apresentado na Tabela 2.

Tabela 2. Parâmetros experimentais por instância.

Parâmetro	51 cidades	100 cidades
Temperatura inicial (T_0)	1000.0	2000.0
Temperatura mínima (T_{\min})	0.0005	0.0005
Iterações máximas	400000	800000
<i>Parâmetros de Reaquecimento</i>		
Temp. de reaquecimento	$0.3 \times T_0$	$0.3 \times T_0$
Límite de estagnação	80000	80000
Taxa de resfriamento (α)	0.95	0.98

4. Resultados Obtidos

4.1. Experimento 2 vs 3: Impacto do Parâmetro SA_{max}

4.1.1. Resultados para 51 Cidades

Tabela 3. Comparação de resultados com diferentes valores de SA_{max} (51 cidades).

Schedule	SA _{max} = 3		SA _{max} = 7	
	Custo	Gap (%)	Custo	Gap (%)
0 (Linear)	548.62	+28.78	498.82	+17.09
5 (Cosseno)	489.42	+14.89	485.64	+14.00
6 (Tang. Hip.)	466.15	+9.43	447.51	+5.05
8 (Exponencial)	481.40	+13.01	456.18	+7.08
9 (Exp. Quad.)	463.21	+8.74	460.44	+8.09

4.1.2. Resultados para 100 Cidades

Tabela 4. Comparação de resultados com diferentes valores de SA_{max} (100 cidades).

Schedule	$SA_{max} = 7$		$SA_{max} = 15$	
	Custo	Gap (%)	Custo	Gap (%)
0 (Linear)	24891.04	+16.96	23709.74	+11.41
5 (Cosseno)	24537.86	+15.30	25773.82	+21.11
6 (Tang. Hip.)	24389.12	+14.60	25263.76	+18.71
8 (Exponencial)	24200.47	+13.72	24821.96	+16.64
9 (Exp. Quad.)	24004.30	+12.80	25625.99	+20.42

4.1.3. Análise

Para 51 cidades, o aumento de SA_{max} de 3 para 7 produziu melhorias consistentes em todos os *schedules*, com o Schedule 6 alcançando gap de apenas 5.05%. Isto evidencia que maior exploração local por nível de temperatura beneficia instâncias menores.

Para 100 cidades, o comportamento foi divergente: apenas o Schedule 0 melhorou significativamente com $SA_{max} = 15$, enquanto os demais pioraram. Este fenômeno sugere que valores excessivos de SA_{max} podem induzir sobre-exploração local prematura em espaços de busca maiores, prejudicando a diversificação do algoritmo. O Schedule Linear, por sua característica de resfriamento uniforme, mostrou-se mais robusto a este efeito.

4.2. Experimento 1 vs 4: Efeito do Reaquecimento com Operador Swap

4.2.1. Resultados para 51 Cidades

Tabela 5. Comparação de resultados com e sem reaquecimento - operador swap (51 cidades).

Schedule	Sem Reaquecimento		Com Reaquecimento	
	Custo	Gap (%)	Custo	Gap (%)
0 (Linear)	498.82	+17.09	493.21	+15.78
5 (Cosseno)	485.64	+14.00	481.72	+13.08
6 (Tang. Hip.)	447.51	+5.05	455.67	+6.96
8 (Exponencial)	456.18	+7.08	461.92	+8.43
9 (Exp. Quad.)	460.44	+8.09	455.83	+7.00

4.2.2. Resultados para 100 Cidades

Tabela 6. Comparação de resultados com e sem reaquecimento - operador swap (100 cidades).

Schedule	Sem Reaquecimento		Com Reaquecimento	
	Custo	Gap (%)	Custo	Gap (%)
0 (Linear)	23709.74	+11.41	23870.42	+12.17
5 (Cosseno)	25773.82	+21.11	24350.49	+14.42
6 (Tang. Hip.)	25263.76	+18.71	23000.87	+8.07
8 (Exponencial)	24821.96	+16.64	24249.71	+13.95
9 (Exp. Quad.)	25625.99	+20.42	23394.59	+9.93

4.2.3. Análise

O reaquecimento apresentou efeitos distintos conforme o tamanho da instância. Para 51 cidades, degradou 2 dos 5 *schedules*, com o Schedule 6 piorando de 5.05% para 6.96%. Isso indica que o mecanismo introduz perturbações desnecessárias em espaços de busca menores, onde a convergência prematura é menos problemática.

Para 100 cidades, o reaquecimento produziu melhorias substanciais em todos os *schedules* exceto o Linear. O Schedule 6 melhorou drasticamente de 18.71% para 8.07%, evidenciando a eficácia do mecanismo em escapar de mínimos locais em espaços maiores. Isso pode ser identificado no gráfico do Schedule 6, onde é alcançado um ponto satisfatório após sair do mínimo local.

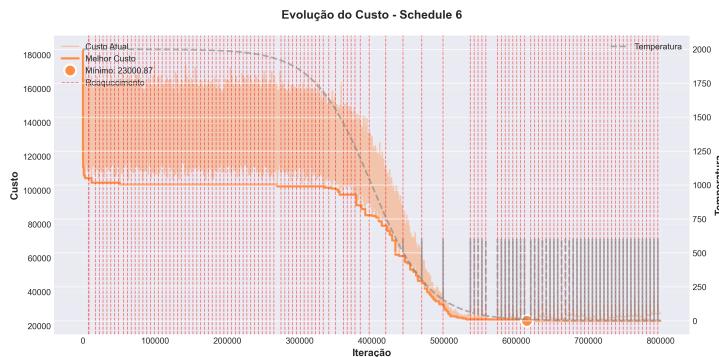


Figura 1. Evolução de custo do Schedule 6 para 51 cidades.

O comportamento anômalo específico do Schedule Linear sugere incompatibilidade entre seu decréscimo uniforme e as reinicializações térmicas. Conclui-se que o reaquecimento deve ser aplicado seletivamente para instâncias grandes ($n \geq 100$) combinado com *schedules* não lineares.

4.3. Experimento 1 vs 5: Comparação Swap vs 2-opt com Reaquecimento

Este experimento comparou o operador *swap* (sem reaquecimento) com a abordagem híbrida 2-opt (com reaquecimento), contrastando a configuração inicial com a final. Foram

executadas 10 execuções independentes (com diferentes seeds, como mencionado na Seção 3) para análise estatística, com foco no desvio padrão dos dados.

4.3.1. Resultados para 51 Cidades

Tabela 7. Comparação entre operadores (10 execuções, 51 cidades).

Schedule	Swap (sem reaquec.)			2-opt (com reaquec.)		
	Mín.	DP	Gap (%)	Mín.	DP	Gap (%)
0 (Linear)	477.04	19.79	+11.97	442.77	8.08	+3.93
5 (Cosseno)	443.60	14.07	+4.13	434.04	5.31	+1.88
6 (Tang. Hip.)	447.51	8.95	+5.05	428.87	3.07	+0.67
8 (Exponencial)	450.09	10.27	+5.65	429.12	2.30	+0.73
9 (Exp. Quad.)	445.19	11.23	+4.50	429.53	3.35	+0.83

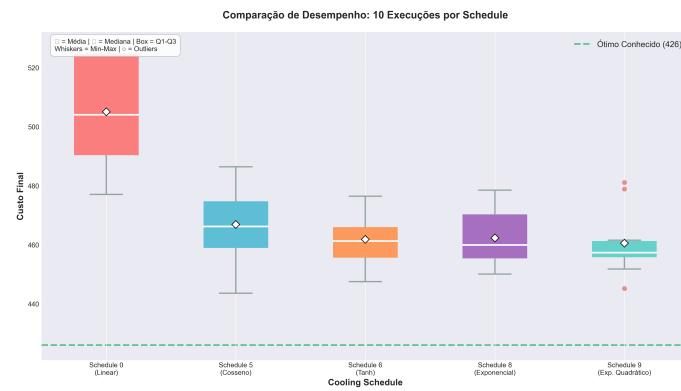


Figura 2. Comparação de desempenho: Operador swap sem reaquecimento (51 cidades).

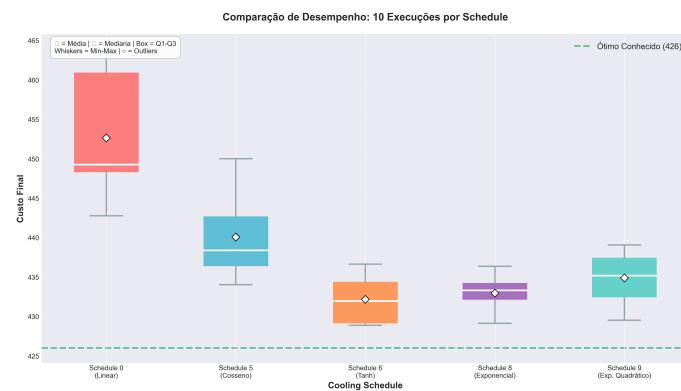


Figura 3. Comparação de desempenho: Operador 2-opt com reaquecimento (51 cidades).

4.3.2. Resultados para 100 Cidades

Tabela 8. Comparação entre operadores (10 execuções, 100 cidades).

Schedule	Swap (sem reaquec.)			2-opt (com reaquec.)		
	Mín.	DP	Gap (%)	Mín.	DP	Gap (%)
0 (Linear)	22727.74	1058.23	+6.79	21370.12	93.88	+0.41
5 (Cosseno)	22874.71	1098.11	+7.48	21362.19	142.05	+0.38
6 (Tang. Hip.)	23364.04	778.55	+9.78	21448.99	93.41	+0.78
8 (Exponencial)	22581.53	926.75	+6.11	21417.47	78.97	+0.64
9 (Exp. Quad.)	23691.89	797.69	+11.32	21344.72	95.54	+0.29

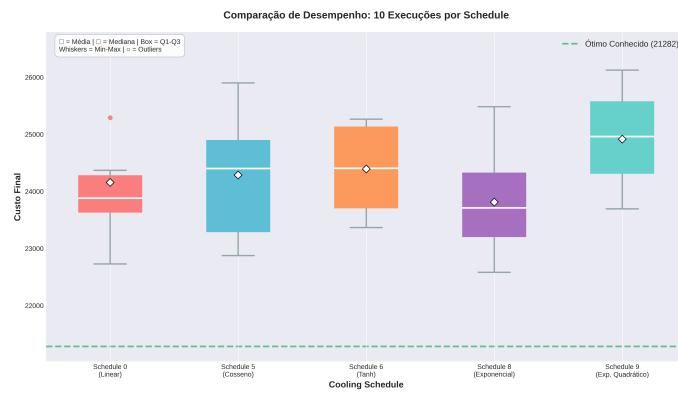


Figura 4. Comparação de desempenho: Operador swap sem reaquecimento (100 cidades).

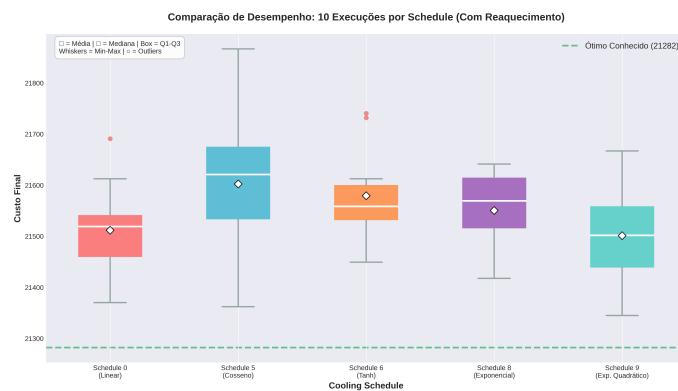


Figura 5. Comparação de desempenho: Operador 2-opt com reaquecimento (100 cidades).

4.3.3. Análise

A combinação do operador 2-opt híbrido com reaquecimento produziu melhorias drásticas em ambas as instâncias. Para 51 cidades, todos os *schedules* apresentaram gaps

inferiores a 2%, com destaque para o Schedule 6 (gap de 0.67%). Além disso, os desvios padrão foram reduzidos significativamente, indicando maior robustez e consistência da abordagem híbrida.

Para 100 cidades, as melhorias foram ainda mais expressivas. Todos os *schedules* alcançaram gaps inferiores a 1%, com o Schedule 9 atingindo gap notável de 0.29%. A redução drástica dos desvios padrão (de aproximadamente 1000 para aproximadamente 90) demonstra que a abordagem híbrida não apenas encontra melhores soluções, mas o faz de forma mais confiável.

Os boxplots ilustram claramente esta superioridade: a distribuição de custos com 2-opt é significativamente mais concentrada e próxima do ótimo, enquanto o *swap* apresenta maior dispersão e outliers. Isto confirma que o operador 2-opt, por eliminar cruzamentos e realizar mudanças estruturais mais significativas, é fundamental para alcançar soluções de alta qualidade em problemas de roteamento.

5. Análise Geral dos Resultados

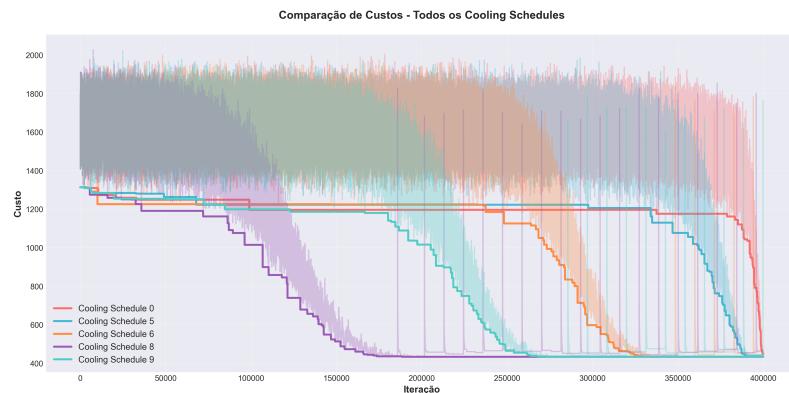


Figura 6. Gráfico de convergência geral: Operador 2-opt com reaquecimento (51 cidades)

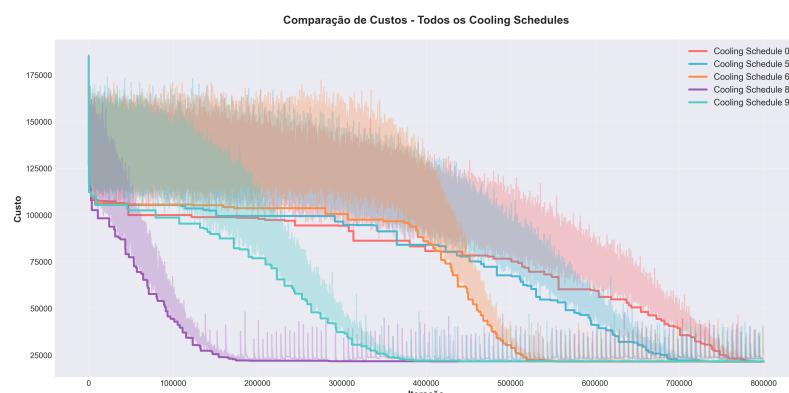


Figura 7. Gráfico de convergência geral: Operador 2-opt com reaquecimento (100 cidades).

Os experimentos realizados revelam padrões consistentes sobre o comportamento do Simulated Annealing para o TSP. O parâmetro SA_{max} demonstrou sensibilidade ao tamanho

da instância, onde valores maiores beneficiam instâncias pequenas mas podem prejudicar instâncias grandes ao induzir sobre-exploração local.

O mecanismo de reaquecimento mostrou-se essencial para instâncias maiores, onde o risco de convergência prematura é maior. Sua eficácia é particularmente pronunciada quando combinado com *schedules* não lineares, que mantêm temperaturas elevadas por períodos mais longos, permitindo melhor aproveitamento das fases de reaquecimento.

A superioridade do operador 2-opt, especialmente quando implementado de forma híbrida, confirma sua importância para problemas de roteamento. Sua capacidade de eliminar cruzamentos e realizar mudanças estruturais significativas na rota é fundamental para escapar de configurações subótimas. A estratégia híbrida (70% 2-opt, 30% *swap*) mostrou-se particularmente eficaz ao equilibrar movimentos de grande e pequena escala.

Entre os *schedules* testados, os não lineares (especialmente Tangente Hiperbólica e Exponencial Quadrático) apresentaram melhor desempenho na configuração avançada, sugerindo que perfis de resfriamento mais suaves facilitam a exploração efetiva do espaço de busca quando combinados com operadores eficientes e reaquecimento adaptativo.

6. Conclusões e Trabalhos Futuros

Este trabalho investigou sistematicamente diferentes configurações do Simulated Annealing para o TSP, avaliando operadores de vizinhança, estratégias de resfriamento e mecanismos de reaquecimento. Os resultados demonstram que a configuração ótima é dependente do tamanho da instância e que a combinação adequada de componentes é crucial para o desempenho do algoritmo.

A configuração final proposta, na qual foram aplicados operador 2-opt híbrido, reaquecimento adaptativo e *schedules* não lineares, alcançou gaps inferiores a 1% para ambas as instâncias testadas, demonstrando a eficácia da abordagem. A redução significativa dos desvios padrão indica que esta configuração é não apenas eficaz, mas também robusta. Porém, algumas melhorias poderiam ser introduzidas: investigações mais profundas sobre valores ideais de SA_{max} para instâncias grandes (como as de 100 cidades), normalização da matriz de distâncias, testes com maior número de iterações, análise mais detalhada da influência das seeds e refinamento de parâmetros.

Como trabalhos futuros, sugere-se: (i) investigação de critérios adaptativos para SA_{max} baseados no tamanho da instância e fase da busca; (ii) exploração de estratégias de reaquecimento com múltiplos níveis de intensidade; (iii) análise de operadores de vizinhança adicionais como 3-opt e Lin-Kernighan; (iv) desenvolvimento de critérios de parada mais sofisticados baseados em métricas de diversidade populacional; e (v) aplicação da metodologia desenvolvida a outras variantes do TSP e problemas de roteamento relacionados.

Referências

- [1] Free Software Foundation. *GNU Scientific Library Reference Manual: Simulated Annealing*. GNU Project, 2025. Acesso em: 18 nov. 2025.
- [2] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by simulated annealing. *Science*, 220(4598):671–680, 1983.