



UNIVERSIDADE FEDERAL DE OURO PRETO
INSTITUTO DE CIÊNCIAS EXATAS E BIOLÓGICAS
DEPARTAMENTO DE ESTATÍSTICA
BACHARELADO EM ESTATÍSTICA



Avaliação de Mecanismo Probabilístico de *Cooling Schedule* para o Algoritmo *Simulated Annealing*

Henrique Dornelas Leijoto

Ouro Preto-MG
2024

Henrique Dornelas Leijoto

**Avaliação de Mecanismo Probabilístico de *Cooling*
Schedule para o Algoritmo *Simulated Annealing***

Monografia de Graduação apresentada ao Departamento de Estatística do Instituto de Ciências Exatas e Biológicas da Universidade Federal de Ouro Preto como requisito parcial para a obtenção do grau de bacharel em Estatística.

Orientador: Anderson Ribeiro Duarte

Ouro Preto

2024

SISBIN - SISTEMA DE BIBLIOTECAS E INFORMAÇÃO

L529a Leijoto, Henrique.

Avaliação de mecanismo probabilístico de Cooling Schedule para o algoritmo Simulated Annealing. [manuscrito] / Henrique Leijoto. - 2024. 33 f.: il.: color., gráf., tab..

Orientador: Prof. Dr. Anderson Duarte.

Monografia (Bacharelado). Universidade Federal de Ouro Preto. Instituto de Ciências Exatas e Biológicas. Graduação em Estatística .

1. Probabilidades. 2. Algoritmos computacionais. 3. Caixeiros-viajantes. 4. Eficácia. I. Duarte, Anderson. II. Universidade Federal de Ouro Preto. III. Título.

CDU 519.2

Bibliotecário(a) Responsável: Paulo Vitor Oliveira - CRB6/2551



FOLHA DE APROVAÇÃO

Henrique Dornelas Leijoto

Avaliação de mecanismo probabilístico de *cooling schedule* para o algoritmo *simulated annealing*

Monografia apresentada ao Curso de Estatística da Universidade Federal de Ouro Preto como requisito parcial para obtenção do título de Bacharel em Estatística

Aprovada em 21 de fevereiro de 2024

Membros da banca

Dr. Anderson Ribeiro Duarte - Orientador (Universidade Federal de Ouro Preto)
Dr. Josino José Barbosa - Membro (Universidade Federal de Ouro Preto)
Dr. Helgem de Souza Ribeiro Martins - Membro (Universidade Federal de Ouro Preto)

Prof. Dr. Anderson Ribeiro Duarte, orientador do trabalho, aprovou a versão final e autorizou seu depósito na Biblioteca Digital de Trabalhos de Conclusão de Curso da UFOP em 21/02/2024



Documento assinado eletronicamente por **Anderson Ribeiro Duarte, PROFESSOR DE MAGISTERIO SUPERIOR**, em 24/02/2024, às 08:23, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



Documento assinado eletronicamente por **Helgem de Souza Ribeiro Martins, PROFESSOR DE MAGISTERIO SUPERIOR**, em 26/02/2024, às 10:16, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



Documento assinado eletronicamente por **Josino José Barbosa, PROFESSOR DE MAGISTERIO SUPERIOR**, em 26/02/2024, às 10:42, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



A autenticidade deste documento pode ser conferida no site http://sei.ufop.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **0673141** e o código CRC **E6E09440**.

Agradecimentos

Gostaria de expressar minha profunda gratidão aos meus pais, cujo apoio e constante incentivo aos estudos foram fundamentais durante meu crescimento pessoal. A dedicação em priorizar a educação em nossa família foi essencial para meu desenvolvimento e para me tornar uma pessoa com gosto por estudos. Aos meus colegas de curso, em especial ao Fernando, Ana, Victor e Paulo Vitor, agradeço pelos anos de parceria, amizade e por tornarem o processo educacional mais fácil e divertido. Aos professores do DEEST e ao meu orientador, Dr. Anderson Ribeiro Duarte, sou extremamente grato pela excelência no ensino e pela atenciosidade que sempre demonstram aos discentes. Por fim, não posso deixar de agradecer à República Exílio, que foi meu lar em Ouro Preto e minha segunda família. As amizades feitas lá serão para toda a vida.

Resumo

Desde da sua concepção na década de 40, os algoritmos de Monte Carlo são utilizados para resolver diversos tipos de problemas, tais como problemas de amostragem e estimação, encontrando aplicações na Física, Biologia e Engenharia. Dentre suas muitas variações, algoritmos de Monte Carlo acoplados a cadeias de Markov (MCMC) estão entre os mais poderosos, tais como Metropolis-Hastings e simulated annealing. Com a crescente quantidade de dados e demanda por eficiência computacional, tais algoritmos vem são como base de técnicas emergentes em Ciências dos Dados e Inteligência Artificial. Algoritmo de Metropolis para Monte Carlo foi nomeado pela revista IEEE Computing in Science & Engineering como um dos 10 algoritmos que mais influenciaram o desenvolvimento e a prática da ciência e engenharia no século 20! O clássico algoritmo *Simulated Annealing* é uma vertente desse algoritmo com eficácia e vasta aplicabilidade em problemas de otimização de interesse prático. Este estudo busca apresentar estas constatações e avaliar a estrutura específica do algoritmo através de testes de adaptabilidade por meio do conhecido problema do caixeiro viajante.

Palavras-chave: Recozimento Simulado, Cronograma de Resfriamento, Problema do Caixeiro Viajante, Eficácia, Eficiência.

Abstract

Since their conception in the 40s, Monte Carlo algorithms have been used to solve different types of problems, such as sampling and estimation problems, with applications in physics, biology, and engineering. Among its many variations, Monte Carlo Markov chains algorithms (MCMC) are among the most powerful, such as Metropolis-Hastings and *Simulated Annealing*. With the growing amount of data and demand for computational efficiency, such algorithms are used as the basis of emerging Data Science and Artificial Intelligence techniques. Metropolis Algorithm for Monte Carlo was named by IEEE Computing in Science & Engineering journal as one of the ten algorithms that most influenced the development and practice of science and engineering in the 20th century. The classic *Simulated Annealing* algorithm is a branch of these algorithms with effectiveness and broad applicability in optimization problems of practical interest. This study seeks to present these findings and evaluate the specific structure of this algorithm through adaptability tests using the well-known traveling salesman problem.

Keywords: Simulated Annealing, Cooling Schedule, Traveling Salesman Problem, Effectiveness, Efficiency.

Lista de ilustrações

Figura 1 – Mudança de Estado com Variação de Temperatura Não Planejada.	5
Figura 2 – Mudança de Estado com Variação de Temperatura Planejada.	5
Figura 3 – Diferentes Estratégias de <i>Cooling Schedules</i>	20
Figura 4 – Diferentes Valores de α para a Estratégia de Resfriamento 1.	21
Figura 5 – Valores Vizinhos ao 0,50 para α com a Estratégia de Resfriamento 1.	22
Figura 6 – Diferentes Valores de γ para a Estratégia de Resfriamento 2.	23
Figura 7 – Valores Vizinhos ao 0,50 para γ com a Estratégia de Resfriamento 2.	24
Figura 8 – Comparação de Resultados para os Esquemas de Redução de Temperatura 1, 2 e 3.	25
Figura 9 – Comparação de Custo Final das Soluções Fornecidas para os Esquemas de Redução de Temperatura 1, 2 e 3 e a Solução Inicial.	26
Figura 10 – Comparação de Resultados para os Esquemas de Redução de Temperatura 1, 2 e 3 por Médias Acumuladas.	27

Sumário

1	INTRODUÇÃO	1
1.1	Motivação	2
1.2	Objetivos	2
1.2.1	Objetivos Gerais	3
1.2.2	Objetivos Específicos	3
2	FUNDAMENTAÇÃO TEÓRICA	5
2.1	Aplicações do Algoritmo <i>Simulated Annealing</i>	6
2.2	Monte Carlo: Uma Abordagem Intuitiva	7
2.3	Cadeias de Markov e a Propriedade Markoviana	8
2.4	A Distribuição Estacionária	9
2.5	Algoritmos MCMC: Uma Visão Geral	9
3	ASPECTOS METODOLÓGICOS	11
3.1	Algoritmos de Busca Local	11
3.2	Algoritmo Metropolis	12
3.3	Algoritmo <i>Simulated Annealing</i>	14
3.4	Definição do Problema do Caixeiro Viajante e suas Suposições	15
3.5	Desafios na Busca da Solução Ótima para o PCV	16
3.6	<i>Simulated Annealing</i> : Uma Solução Eficiente e Robusta	17
4	EXPERIMENTOS COMPUTACIONAIS	19
4.1	Calibração das Estratégias de Resfriamento 1 e 2	20
4.2	Comparação entre as Três Estratégias de Resfriamento	24
5	CONSIDERAÇÕES FINAIS	29
5.1	Propostas para Trabalhos Futuros	30
	REFERÊNCIAS	31

1 Introdução

A otimização é um processo de maximização ou minimização de alguma função. Usualmente essas funções podem envolver uma única variável, ou até mesmo, várias variáveis e resultar em uma solução ótima, que é a melhor solução, ou um conjunto de soluções ótimas para algum problema específico de interesse. Além disso, a otimização pode ser encontrada em diversas áreas da Ciência e Tecnologia, e tem um papel crucial para que processos que são muitas vezes onerosos e/ou demasiadamente trabalhosos sejam mais eficientes no dia a dia daqueles que os utilizam.

Dessa maneira, a otimização desempenha um papel relevante em diversas esferas de interesse prático. Abrangendo desde a eficiência na utilização de recursos, como tempo, dinheiro, materiais e mão de obra, até a minimização de custos, consumo de recursos, tempo de execução e aprimoramento da qualidade na produção, em diversos setores, como por exemplo: industrial, médico, logístico, serviços, entre tantos outros. A crescente competitividade do mercado acentua a importância de otimização de processos, especialmente na logística e transporte, em que o planejamento de rotas e estratégias pode resultar em economia de tempo e redução de consumo de combustível, além de aumentar a eficácia operacional. Em resumo, a otimização é uma poderosa ferramenta para estruturar processos em diversos setores da Indústria e Tecnologia, capaz de permitir tomadas de decisão conscientes e melhorias no desempenho de métodos anteriormente não otimizados.

É possível encontrar a utilização de estratégias de otimização em várias áreas, particularmente a Estatística está bem presente neste contexto. Um exemplo recorrente pode ser visualizado no método da máxima verossimilhança, método bastante difundido em Estatística, pois possui sua principal função em fornecer os estimadores que maximizam a função de verossimilhança, ou seja, os estimadores capazes de fornecer estimativas mais verossímeis para o valor do parâmetro a ser estimado. Além desse caso, tem-se o método dos mínimos quadrados ordinários e mínimos quadrados ponderados, de vasta utilização em estudos de Análise de Regressão e também em algumas outras técnicas de modelagem. Essa metodologia busca fornecer estimadores de parâmetros que sejam capazes de minimizar a soma dos quadrados entre valores observados e valores previstos.

Apesar das técnicas de otimização serem importantes e de grande impacto no dia a dia no campo da Ciência e Tecnologia, alguns problemas de grande dificuldade podem ser encontrados, como: alta complexidade da função objetivo; presença de um grande número de restrições; demanda computacional alta. Por isso, existe a necessidade de

estratégias de fornecimento de soluções que apresentem resultados satisfatórios, e que ao mesmo tempo não sejam demasiadamente onerosas em sua utilização, por exemplo em consumo de tempo de processamento computacional.

Para enfrentar os desafios como os mencionados anteriormente, os métodos heurísticos e meta-heurísticos surgem como uma excelente alternativa. Esses métodos têm a habilidade necessária para fornecer soluções que, embora não sejam necessariamente as ótimas, são soluções ditas sub-ótimas, mas de grande eficiência, e não exigem um grande esforço computacional. Essas abordagens utilizam regras, melhorias iterativas e algoritmos que exploram eficientemente o espaço de soluções de forma rápida e abrangente. Ao contrário dos métodos exatos, as heurísticas visam alcançar resultados factíveis e satisfatórios em contraponto de soluções ótimas exatas, muitas vezes inviáveis de serem alcançadas. Este trabalho, abordará a clássica técnica heurística conhecida pelo termo do inglês *Simulated Annealing*, ou em uma tradução imediata, Recozimento Simulado.

1.1 Motivação

A natureza probabilística do *Simulated Annealing* é o que o torna particularmente atraente e motivador para estudos aprofundados nas áreas Matemática e Estatística. O algoritmo faz uso de uma função de aceitação baseada em probabilidades para decidir se uma nova solução proposta deverá ser aceita, mesmo que essa nova solução seja inferior à solução atual. Esse aspecto probabilístico proporciona flexibilidade e capacidade de explorar o espaço de busca de maneira mais abrangente. Esse caráter estocástico, em conjunto com a capacidade de escapar de mínimos locais, torna o *Simulated Annealing* um candidato valioso para uma gama diversificada de problemas complexos de otimização. Particularmente, o *Simulated Annealing* se enquadra na classe dos algoritmos de Monte Carlo via Cadeias de Markov, ou mais usualmente do inglês, *Markov chain Monte Carlo* (MCMC). Portanto, uma análise detalhada das condições, das estratégias de implementação e particularmente do mecanismo probabilístico de aceitação são essenciais para extrair todo o potencial e a eficácia desse algoritmo em diferentes contextos aplicados.

1.2 Objetivos

Este estudo tem objetivos de caráter científico e aplicado. As investigações sobre os métodos MCMC tem interesse científico explícito em estudos de diversas áreas, particularmente em mecanismos de otimização e estratégias Bayesianas. Por outro lado, a modelagem de problemas de otimização nas mais variadas áreas e um ambiente

de aplicação para técnicas e algoritmos de otimização como o algoritmo *Simulated Annealing*.

1.2.1 **Objetivos Gerais**

- i. discutir os aspectos de concepção e utilização da heurística *Simulated Annealing*;
- ii. obter uma implementação capaz de fornecer soluções factíveis para problemas clássicos de otimização;
- iii. estabelecer comparações em variações das estratégias de probabilidade de aceitação inerentes ao algoritmo *Simulated Annealing*.

1.2.2 **Objetivos Específicos**

- i. apresentar uma revisão bibliográfica que direcione para os assuntos da proposição e utilização do algoritmo *Simulated Annealing*;
- ii. discutir os resultados comparativos entre diferentes formulações para a probabilidade de aceitação inerente ao algoritmo *Simulated Annealing*.

Este texto é organizado da seguinte forma, inicialmente um capítulo introdutório que aborda aspectos de pesquisa discutidos durante a concepção desse estudo, bem como o delineamento prévio de objetivos a serem trabalhados. Em seguida, o segundo capítulo apresenta uma relevante revisão conceitual e bibliografia sobre esse tema de pesquisa. A seguir, o capítulo de Aspectos Metodológicos detalha o algoritmo *Simulated Annealing*. O quarto capítulo apresenta de forma mais detalhada todo o conjunto de resultados alcançados para as variações propostas na probabilidade de aceitação. Por fim, o último capítulo apresenta as conclusões alcançadas por meio dessa investigação e também propostas de continuidade desse estudo.

2 Fundamentação Teórica

O algoritmo *Simulated Annealing* é um método metaheurístico bastante simples e extremamente difundido para resolver problemas de otimização difíceis, com funções objetivo não explícitas e que usualmente exigem alguma simulação computacional onerosa. A expressão “*simulated annealing*” fornece mais de um milhão de acessos quando pesquisada no mecanismo de busca Google Acadêmico dedicado à literatura acadêmica.

No início da década de 1980, três pesquisadores vinculados à gigante da informática IBM, Kirkpatrick, Gelatt e Vecchi (1983) [1], introduziram o algoritmo *Simulated Annealing* na otimização combinatória. Os conceitos eram baseados em uma analogia com o recozimento físico de materiais. No processo de recozimento, um sólido é levado para um estado de baixa energia após aumentar sua temperatura. Em paralelo, Černý (1985) [2] também trabalhava em versão similar para o propósito, as duas versões foram naturalmente unificadas no meio acadêmico.

O procedimento pode ser resumido em duas etapas, levar o sólido a uma temperatura muito elevada até “derreter” a sua estrutura. Posteriormente resfriar o sólido de acordo com um esquema de diminuição de temperatura muito particular para atingir um estado sólido de energia mínima. Em fase líquida, as partículas do material são distribuídas de forma aleatória. As Figuras 1 e 2 resumem o funcionamento esquemático desse procedimento físico.

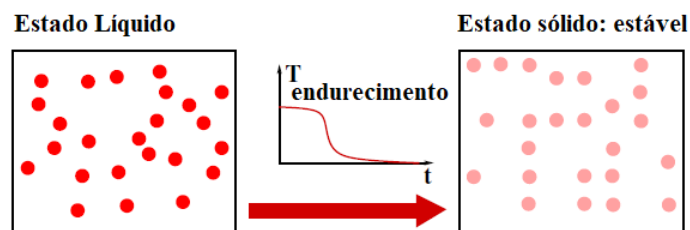


Figura 1 – Mudança de Estado com Variação de Temperatura Não Planejada.

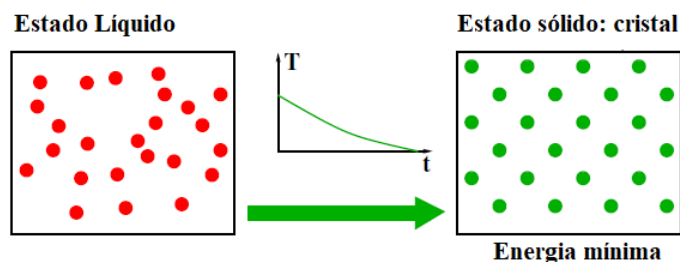


Figura 2 – Mudança de Estado com Variação de Temperatura Planejada.

Para que o estado de energia mínima seja alcançado é necessária uma temperatura inicial que seja suficientemente alta e um tempo de resfriamento que seja suficientemente longo. Caso contrário, o sólido será encontrado em estado estável com energia não mínima; isso é chamado de endurecimento, que consiste no resfriamento repentino de um sólido.

2.1 Aplicações do Algoritmo *Simulated Annealing*

O algoritmo *Simulated Annealing* é reconhecido por sua versatilidade e aplicabilidade em uma ampla gama de domínios. Algumas das áreas em que o algoritmo *Simulated Annealing* é amplamente utilizado incluem:

- projetos de engenharia com otimização de *layouts* de circuitos integrados, roteamento de fios, projeto de redes [3–5];
- logística e transporte com otimização de rotas de entrega, escalonamento de tarefas, planejamento de horários [6–8];
- bioinformática com o dobramento de proteínas, *design* de sequências de DNA, análise de dados biológicos [9, 10];
- procesos de manufatura com a otimização de processos de fabricação, programação de máquinas, controle de qualidade [11, 12];
- problemas combinatórios gerais na resolução de problemas NP-difíceis, como o clássico problema do caixeiro-viajante [13, 14];
- *machine learning* com a otimização de hiperparâmetros em algoritmos de aprendizado de máquina [15–17];
- engenharia de *software* com a otimização de *design* de *software*, alocação de recursos, agendamento de projetos [18, 19];
- teoria de filas com a otimização de consumo de recursos e a eficiência do atendimento de redes de filas [20–22];

entre diversas outras possibilidades de aplicação.

O *Simulated Annealing* oferece uma abordagem flexível para resolver problemas de otimização complexos e é particularmente eficaz quando outras técnicas não conseguem encontrar soluções viáveis. De posse dessa constatação, o objetivo desse estudo será descrever o algoritmo *Simulated Annealing* para otimização. Entretanto, vale ressaltar que o algoritmo utiliza princípios dos algoritmos MCMC e é uma extensão

dos algoritmos de otimização de busca local. Diante disso, é necessário apresentar conceitualmente os princípios associados aos métodos de Monte Carlo.

2.2 Monte Carlo: Uma Abordagem Intuitiva

Os fundamentos matemáticos e uma visão intuitiva da forma mais geral das técnicas de Monte Carlo serão abordadas. O conceito da aleatoriedade e seu papel na obtenção de amostras representativas serão explorados.

A técnica de Monte Carlo é fundamentada no uso da aleatoriedade para abordar problemas complexos. Em vez de buscar soluções determinísticas, introduz-se aleatoriedade de maneira controlada. A intuição subjacente é que, ao gerar um grande número de pontos aleatórios dentro do espaço de interesse, é possível usar a proporção de pontos pertencentes a uma região específica para estimar a área ou o valor de uma função nessa região.

Por exemplo, considere a integração de uma função $f(x)$ que seria computacionalmente onerosa. Em vez de calcular a integral de forma analítica, um conjunto de valores aleatórios dentro do domínio da função é gerado. Em seguida, a quantidade desses valores que estão localizados abaixo da curva da função. A razão de pontos abaixo da curva em relação ao número total de pontos gerados fornece uma estimativa do valor da integral definida da função naquela região. Quanto maior o número de valores gerados, mais precisa é a estimativa.

Para ilustrar essa ideia, pode-se utilizar a seguinte forma para representar a estimativa de uma integral por Monte Carlo:

$$\int_a^b f(x)dx \approx \frac{n}{N}(b-a), \quad (2.1)$$

em que a e b representam os limites de integração, $f(x)$ é a função a ser integrada, n é o número de valores aleatórios posicionados abaixo da curva da função $f(x)$, N é o número total de valores aleatórios gerados e $(b-a)$ representa a largura do intervalo de integração. Essa fórmula ilustra como o método Monte Carlo pode ser aplicado para aproximar uma integral.

As técnicas de Monte Carlo são estendidas para variados problemas que envolvem a necessidade de simulação de variáveis aleatórias. Particularmente, os procedimentos que reproduzem por simulação a sequência de passos de cadeias markovianas estão inseridas neste contexto e são pano de fundo para a nomenclatura algoritmos MCMC. Para introduzir este assunto, a definição de Processo Estocástico e a evolução do conceito até o caso específico das Cadeias de Markov deve ser apresentada.

Um processo estocástico é uma coleção de variáveis aleatórias indexadas $X = \{X_t; t \in T\}$ em que T é dito espaço de parâmetros, com $T = [0, \infty)$ ou $T = \{0, 1, 2, \dots\}$. No primeiro caso, X é um processo a tempo contínuo, no segundo X é um processo a tempo discreto. Neste contexto, todas as variáveis aleatórias da família X estão definidas no mesmo espaço amostral. O contra domínio dessas variáveis aleatórias é denominado espaço de estados E . Se E é discreto, o processo estocástico é dito discreto. Se E é contínuo, o processo estocástico é dito contínuo. A abordagem de Cadeias de Markov está particularmente focada em processos estocásticos discretos à tempo discreto.

2.3 Cadeias de Markov e a Propriedade Markoviana

A propriedade markoviana é uma propriedade inerente aos processos estocásticos, entretanto não satisfeita para quaisquer procesos estocásticos. Um processo estocástico atende a propriedade de Markov se é possível afirmar que o futuro não depende do passado se o presente é conhecido. Um processo estocástico a tempo discreto com espaço de estados finito satisfazendo a propriedade de Markov é dito ser uma cadeia de Markov. Matematicamente, a propriedade markoviana é definida da seguinte maneira:

$$P(X_{n+1} = x_{n+1} | X_n = x_n, X_{n-1} = x_{n-1}, \dots, X_0 = x_0) = P(X_{n+1} = x_{n+1} | X_n = x_n) \quad (2.2)$$

Essa equação afirma que a probabilidade de estar em um estado futuro x_{n+1} depende apenas do estado atual x_n . Isso implica que o conhecimento do histórico completo x_0, x_1, \dots, x_n não é necessário para prever o futuro, apenas o último estado visitado x_n é relevante.

Suponha um sistema que pode a cada instante temporal (tempo discretizado) permanecer em algum estado, cada um destes estados representa uma situação distinta do sistema, e as transições entre esses estados ocorrem com base em alguma regra ou probabilidade. Por exemplo, considere um jogo de xadrez, em que cada estado representa uma configuração do tabuleiro, e as transições ocorrem à medida que os jogadores fazem movimentos.

A ideia fundamental das Cadeias de Markov é modelar sistemas como esse, em que a transição de um estado para outro é incerta e depende apenas do estado atual, independentemente do histórico anterior. A regra de transição em uma Cadeia de Markov é determinada por uma matriz de transições. Para uma cadeia de Markov com espaço de estados composto por m estados, uma matriz $m \times m$ representa estas transições, aqui essa matriz será sempre denotada por matriz P .

Dentre diversas propriedades relevantes associadas às cadeias de Markov, é possível elencar a irredutibilidade, a periodicidade, a recorrência e a recorrência positiva, entre outras. O clássico livro de Bhattacharya e Waymire (2009) [23] detalha bem essas propriedades.

2.4 A Distribuição Estacionária

Para toda Cadeia de Markov que seja irredutível, com espaço de estados E finito e com algum estado que seja aperiódico, essa cadeia é dita uma Cadeia de Markov **fortemente ergódica**.

Para uma Cadeia de Markov fortemente ergódica, existe a distribuição estacionária π dessa cadeia. Considere um espaço de estado com m estados distintos, a distribuição estacionária é dada por $\pi = (\pi_1, \pi_2, \dots, \pi_m)$ tal que $\lim_{n \rightarrow \infty} p_{ij}^{(n)} = \pi_j$, em longo prazo, π_j é a fração de tempo que a cadeia permanece no estado j . Note que $p_{ij}^{(n)}$ é o elemento da i -ésima linha e j -ésima coluna da matriz $P^{(n)}$.

Seja ℓ uma linha qualquer da matriz $\lim_{n \rightarrow \infty} P^{(n)}$ em um caso de Cadeia de Markov fortemente ergódica, então $\ell = \pi$. Através das conhecidas equações de Chapman-Komolgorov é possível verificar que $P^{(n)} = P^{(n-1)}P$. Diante disso é possível verificar que:

$$\lim_{n \rightarrow \infty} P^{(n)} = \lim_{n \rightarrow \infty} P^{(n-1)}P = \left[\lim_{n \rightarrow \infty} P^{(n-1)} \right] P = \left[\lim_{n \rightarrow \infty} P^{(n)} \right] P.$$

Portanto $\pi = \pi P$ e ainda, $\sum_j \pi_j = 1$. Esse sistema de equações fornece o vetor π (distribuição estacionária). A distribuição estacionária é um estado especial em uma Cadeia de Markov no qual determina a probabilidade da cadeia permanecer nesse estado a longo prazo. Os algoritmos MCMC tem uma associação direta com a convergência associada à distribuição estacionária.

2.5 Algoritmos MCMC: Uma Visão Geral

No campo da Estatística e Computação Científica, o desenvolvimento de algoritmos MCMC tem desempenhado um papel essencial. O objetivo momentâneo dessa revisão conceitual é unificar os conceitos discutidos anteriormente. Os métodos de Monte Carlo são ferramentas eficazes em procedimentos de simulação de valores aleatórios. Por outro lado, as cadeias markovianas descrevem processos estocásticos bastante específicos, cujas transições são regidas pela propriedade markoviana.

Os Algoritmos MCMC unem esses dois conceitos de maneira engenhosa. Eles operam por meio de cadeias de Markov nas quais os estados representam os valores

amostrais aos quais a metodologia busca simular. A propriedade markoviana assegura que as transições de um estado para outro dependem apenas do estado atual. O procedimento usual segue os seguintes passos:

1. **inicialização** - o algoritmo parte de algum estado inicial;
2. **transições markovianas** - a cadeia de Markov realiza transições de um estado para outro com base em uma matriz de transição que depende apenas do estado atual;
3. **geração de amostras** - à medida que a cadeia de Markov visita estados, o procedimento gera realizações amostrais de visitas aos estados, essas realizações são usadas para estimar parâmetros de interesse;
4. **convergência para a distribuição estacionária** - para um volume elevado de execuções, o procedimento conduz à convergência para a distribuição estacionária;
5. **amostragem da distribuição estacionária** - por fim, a amostra gerada é na verdade uma realização simulada da distribuição estacionária, e permite estimar parâmetros e realizar inferências estatísticas.

Os Algoritmos MCMC são algoritmos poderosos que permitem amostrar de distribuições complexas e de alta dimensionalidade. Eles são especialmente úteis em problemas de Inferência Estatística, como a estimativa de parâmetros em modelos probabilísticos complexos. A capacidade de aproximar a distribuição estacionária torna os Algoritmos MCMC uma ferramenta valiosa em muitas áreas de estudo, entre elas Estatística, Ciência da Computação e Ciência de Dados.

3 Aspectos Metodológicos

O algoritmo *Simulated Annealing* é uma técnica de otimização que se enquadra na categoria dos algoritmos MCMC. Como mencionado anteriormente, o algoritmo é inspirado no processo físico do recozimento de materiais, em que um sólido é aquecido e depois resfriado lentamente para alcançar um estado de energia mínimo, ou seja, um estado de maior estabilidade. O objetivo é encontrar uma configuração que minimize uma função de custo, mas em um espaço de busca potencialmente complexo.

A principal ideia por trás do funcionamento do algoritmo é explorar o espaço de soluções de maneira probabilística. Por meio de movimentos que possam aumentar ou diminuir o valor da função de custo. A probabilidade de aceitar um movimento que piora a solução é controlada por um parâmetro de temperatura. Inicialmente, a temperatura é alta, o que permite movimentos mais arriscados, mas ao longo do tempo, a temperatura é reduzida gradualmente, diminuindo a probabilidade de aceitar movimentos desvantajosos. Esse processo simula o resfriamento lento do material no recozimento e ajuda o algoritmo a escapar de mínimos locais e a convergir para soluções próximas do ótimo global.

Uma vez estabelecida a estrutura conceitual dos algoritmos MCMC é importante abordar particularmente o algoritmo alvo desse estudo. Uma apresentação metodológica do clássico algoritmo *Simulated Annealing* é relevante para posicionar este algoritmo na classe de algoritmos MCMC. Para iniciar essa apresentação, é necessário detalhar os princípios dos algoritmos de otimização de busca local.

3.1 Algoritmos de Busca Local

Esses algoritmos otimizam uma função de custo explorando a vizinhança do ponto atual no espaço de solução. Considere um problema geral de otimização com função objetivo de minimização f e seja \mathcal{S} o conjunto de todas as soluções viáveis para o referido problema.

Definição 1. *Seja \mathcal{N} uma aplicação que define para cada solução $i \in \mathcal{S}$, e seja \mathcal{S}_i um subconjunto tal que $\mathcal{S}_i \subset \mathcal{S}$ composto por soluções “vizinhas” (a definir pelo usuário de acordo com o problema de interesse) à solução i . O subconjunto \mathcal{S}_i é chamado de vizinhança da solução i .*

Nas definições seguintes, \mathcal{N} será sempre considerado como uma estrutura de vizinhança associada ao problema de otimização delimitado por f e \mathcal{S} .

Definição 2. Um mecanismo gerador é um método para selecionar uma solução j em qualquer vizinhança \mathcal{S}_i de uma dada solução i .

Um algoritmo de busca local é um algoritmo iterativo que inicia sua busca a partir de um ponto viável, sorteado aleatoriamente no espaço de soluções viáveis. Um mecanismo de geração é então aplicado sucessivamente para encontrar uma solução superior (em termos do valor da função objetivo), explorando a vizinhança da solução corrente. Se tal solução for encontrada, ela se tornará a solução corrente.

O algoritmo termina sua execução quando nenhuma melhoria for encontrada, e a solução atual é considerada como a solução ótima aproximada do problema de otimização.

Definição 3. Uma solução $i^* \in \mathcal{S}$ é dita ser uma solução ótima local em relação a \mathcal{N} para o problema de otimização delimitado por f e \mathcal{S} se $f(i^*) \leq f(j)$, para todo $j \in \mathcal{S}_{i^*}$.

Definição 4. A estrutura de vizinhança \mathcal{N} é dita exata se, para cada ótimo local em relação a \mathcal{N} , $i^* \in \mathcal{S}$, i^* também é um ótimo global para o problema de otimização delimitado por f e \mathcal{S} .

Por definição, os algoritmos de busca local convergem para ótimos locais, a menos que se tenha uma estrutura de vizinhança exata. Esta noção de vizinhança exata é teórica porque geralmente leva, na prática, a recorrer a uma enumeração completa do espaço de busca.

Intuitivamente, se a solução penetrar em um subdomínio sobre o qual a função objetivo é convexa, o algoritmo permanece preso neste subdomínio, a menos que a estrutura de vizinhança associada ao mecanismo de geração possa alcançar pontos fora deste subdomínio.

Para evitar ficar preso em mínimos locais, é então necessário definir um processo que provavelmente aceitará transições de estado atual que reduzem momentaneamente o desempenho (em termos de objetivo) da solução atual: este é o princípio fundamental do algoritmo *Simulated Annealing*. Para descrever o algoritmo, ainda é necessário apresentar o algoritmo Metropolis [24] que é um componente básico do *Simulated Annealing*.

3.2 Algoritmo Metropolis

Em 1953, Metropolis, Rosenbluth e Teller [24] desenvolveram um algoritmo para simular o recozimento físico, conforme descrito anteriormente. O interesse era reproduzir fielmente a evolução da estrutura física de um material submetido à técnica de recozimento. Este algoritmo é baseado em técnicas de Monte Carlo que consistem em gerar uma sequência de estados do sólido.

Em estudos de Estatística e Física Estatística, o algoritmo Metropolis é um método da classe MCMC para obtenção de amostras aleatórias decorrentes de uma distribuição de probabilidade em que a amostragem direta é considerada de difícil execução. Essas amostras geradas podem ser usadas para aproximar a distribuição correlacionada ou para até mesmo calcular uma integral, como por exemplo, o cálculo de um valor esperado. Os algoritmos MCMC são geralmente utilizados para amostragem de distribuições multidimensionais, especialmente nos casos de alta dimensionalidade. Para distribuições unidimensionais, existem diversos outros métodos mais eficientes como a rejeição adaptativa e o uso do Teorema da Transformação Inversa que podem retornar amostras independentes, e escapam do problema do efeito de autocorrelação contido nas amostras obtidas através dos métodos MCMC.

O ponto de partida é um estado inicial i de energia E_i , um novo estado j de energia E_j é gerado pela modificação da posição de uma partícula. Se o novo estado apresenta menor energia, o estado j se torna o novo estado atual. Por outro lado, se o novo estado apresenta maior energia, a diferença de energia será utilizada na determinação de uma probabilidade de aceitação para o novo estado dada por:

$$P(\text{estado corrente ser o estado } j) = \exp\left(\frac{E_i - E_j}{k_b T}\right), \quad (3.1)$$

em que o valor T representa a temperatura do sólido e k_b é a constante de Boltzmann ($k_b = 1,38 \times 10^{-23}$ joule/Kelvin).

Este critério de aceitação do novo estado é usualmente denominado critério Metropolis. Se o resfriamento for realizado de forma suficientemente lenta, o sólido atinge um estado de equilíbrio em cada temperatura T dada. No algoritmo Metropolis, esse equilíbrio é alcançado por meio de um grande número de transições para cada temperatura.

O equilíbrio térmico é caracterizado pela distribuição estatística de Boltzmann. Esta distribuição dá a probabilidade de que o sólido esteja no estado i de energia E_i à temperatura T :

$$P(X = i) = \frac{1}{Z(T)} \exp\left(-\frac{E_i}{k_b T}\right), \quad (3.2)$$

em que X é uma variável aleatória associada ao estado atual do sólido e $Z(T)$ é um coeficiente de normalização, definido por $Z(T) = \sum_{j \in \mathcal{S}} \exp\left(-\frac{E_j}{k_b T}\right)$.

3.3 Algoritmo *Simulated Annealing*

No algoritmo *Simulated Annealing*, o algoritmo Metropolis é aplicado para gerar uma sequência de soluções no espaço de estados \mathcal{S} . O procedimento faz uma analogia entre um sistema multipartículas e o problema de otimização por meio das seguintes equivalências:

- os pontos do espaço de soluções representam os possíveis estados do sólido;
- a função a ser minimizada representa a energia do sólido.

Um parâmetro para controlar a temperatura é introduzido. Além disso, pressupõe-se que a cada ponto do espaço de soluções, uma vizinhança e um mecanismo para obtenção de uma solução nesta vizinhança esteja definido. Por fim, o princípio de aceitação é definido.

Definição 5. *Considere um problema geral de otimização com função objetivo de minimização f e seja \mathcal{S} o conjunto de todas as soluções viáveis para o referido problema. Para i e j dois pontos do espaço de soluções \mathcal{S} , o critério para aceitar a solução j da solução atual i é dado pela seguinte probabilidade:*

$$P(\text{aceitar a solução } j) = \begin{cases} 1 & \text{se } f(j) < f(i) \\ \exp\left(\frac{f(i) - f(j)}{c}\right) & \text{caso contrário} \end{cases} \quad (3.3)$$

em que c é o parâmetro de controle da temperatura. Por analogia, o princípio de geração de um vizinho corresponde ao mecanismo de perturbação do algoritmo Metropolis e o princípio de aceitação representa o critério Metropolis.

Definição 6. *Uma transição representa a substituição da solução atual por uma geração*

- geração;
- aceitação.

Na sequência, seja T_i o valor do parâmetro de temperatura e i seja o número de iterações executadas. O princípio do algoritmo *Simulated Annealing* pode ser resumido pelo Algoritmo 1.

Algoritmo 1: Pseudo-código para o algoritmo *Simulated Annealing*.

```

/* inicialize quantidade de soluções iniciais  $w$  */
/* inicialize a temperatura  $T_1$  */
/* gere um conjunto de soluções iniciais  $\{S_1, S_2, \dots, S_w\}$  */
for  $j = 1; j \leq w, j++$  do
   $i \leftarrow 1$ 
  while  $T_i > 0$  do
    if  $i = 1$  then
      |  $\text{sol\_max} \leftarrow \text{sol}_i \leftarrow C_j$ 
    end
     $\mathcal{U} \leftarrow \text{uniform}(0,1)$ 
    /* utiliza  $\mathcal{U}$  para escolher o critério de vizinhança */
     $\text{sol\_aux} \leftarrow \text{PerturbSolution}(\text{sol}_i)$ 
    if  $\text{sol\_aux}$  é superior a  $\text{sol}_i$  then
      |  $\text{sol}_{i+1} \leftarrow \text{sol\_aux}$ 
      | if  $\text{sol\_aux}$  é superior a  $\text{sol\_max}$  then
      | |  $\text{sol\_max} \leftarrow \text{sol\_aux}$ 
      end
    else
      |  $\mathcal{U} \leftarrow \text{uniform}(0,1)$ 
      |  $\Delta \leftarrow |F(\text{sol}_i) - F(\text{sol\_aux})|$ 
      | if  $\mathcal{U} < e^{-\Delta/T_i}$  then
      | end
      |  $\text{sol}_{i+1} \leftarrow \text{sol\_aux}$ 
    end
     $i \leftarrow i + 1$ 
    /* atualize a temperatura,  $T_i$  */
  end
   $\text{sol\_final}_j \leftarrow \text{sol\_max}$ 
end
sol_final

```

O estudo que será apresentado tem por objetivo estabelecer testes de aplicabilidade em diferentes estratégias para o algoritmo *Simulated Annealing*. Para tanto utilizaremos o conhecidíssimo problema do caixeiro viajante.

3.4 Definição do Problema do Caixeiro Viajante e suas Suposições

O Problema do Caixeiro Viajante (PCV), do inglês *Travelling Salesman Problem* (TSP), é um problema bastante clássico nos estudos de otimização combinatória. Originariamente o problema é uma analogia para o problema real de um vendedor viajante (o caixeiro) que intenciona realizar uma viagem única com intuito de visitar um conjunto pré-determinado de cidades e retornar à cidade de origem ao final do percurso. Além disso, por razões econômicas, o caixeiro busca realizar tal tarefa através do menor percurso possível.

Ao considerar essa analogia, o problema parece demasiadamente simplista e até ingênuo, porém trata-se de um dos problemas mais intensamente investigados em estudos de Matemática Computacional. O nível de aplicabilidade das analogias desse problema é vasto, pode ser aplicado em diversas áreas como logística, genética, manufatura, telecomunicações e neurociência (Cook et al., 2011 [25]).

O PCV é um desafio clássico também em teoria dos grafos e otimização combinatória. Este problema é comumente abordado com as seguintes suposições:

1. **grafo completo** - assume-se a existência de uma conexão direta entre cada par de cidades, formando um grafo completo, isso implica que é possível viajar diretamente de qualquer cidade para qualquer outra cidade;
2. **distância simétrica** - no contexto do PCV Simétrico, a distância entre duas cidades é a mesma, independentemente da ordem em que são percorridas, essa simetria simplifica algumas abordagens de resolução, mas nem sempre é uma característica aplicável.
3. **desigualdade triangular** - idealmente, a desigualdade triangular é satisfeita, o que significa que a distância direta entre duas cidades é sempre menor ou igual à soma das distâncias das rotas que passam por uma terceira cidade, essa propriedade pode facilitar algumas estratégias de otimização.

3.5 Desafios na Busca da Solução Ótima para o PCV

A busca pela solução ótima do PCV é substancialmente desafiadora devido à natureza combinatória do problema. O aumento no número de cidades (vértices) amplia exponencialmente a quantidade de soluções possíveis. Esse nível de complexidade é evidenciado pelos dados expostos na Tabela 1, que compara o número de caminhos possíveis com o tempo estimado para enumerar todas as soluções.

Vértices ou cidades	Número de caminhos	Tempo estimado para enumerar as soluções
5	12	12 μ seg.
10	181,440	0,18 seg.
15	$4,359 \times 10^{10}$	12 hrs.
20	$6,082 \times 10^{16}$	1,928 anos
61	$4,160 \times 10^{81}$	$1,319 \times 10^{67}$ anos

Tabela 1 – Número de caminhos e tempo estimado para solucionar o PCV, adaptado de Hoffman, Padberg e Rinaldi (2003) [26].

O número de caminhos é comparável ao número estimado de átomos no universo visível (10^{80}), e o tempo necessário para enumerar todas as soluções é semelhante à

idade do universo ($13,798 \pm 0,037 \times 10^9$ anos). Isso destaca a impraticabilidade de uma abordagem exaustiva na busca pela solução ótima para o PCV.

3.6 *Simulated Annealing: Uma Solução Eficiente e Robusta*

O *Simulated Annealing* destaca-se como uma abordagem eficaz para o PCV devido à sua natureza probabilística e capacidade de escapar de mínimos locais. Ao introduzir elementos de aleatoriedade e implementar um processo de resfriamento controlado, o algoritmo pode explorar o espaço de soluções de maneira global. Isso permite transições para soluções inicialmente piores. Essa característica é crucial para evitar que o algoritmo fique preso em mínimos locais sub-ótimos, o que é uma limitação comum em abordagens determinísticas.

A dinâmica de resfriamento gradual do *Simulated Annealing* é particularmente vantajosa. Inicialmente, permite movimentos mais amplos e, à medida que a “temperatura” diminui, direciona o algoritmo para soluções cada vez melhores. Essa estratégia, inspirada no processo de resfriamento controlado de metais, confere ao *Simulated Annealing* uma capacidade única de encontrar soluções de alta qualidade em cenários complexos e altamente dimensionalizados, tornando-o uma escolha poderosa para enfrentar o desafiador Problema do Caixeiro Viajante.

Vale ressaltar que o foco dessa investigação reside em discutir diferentes estratégias para essa dinâmica de resfriamento gradual. O caráter probabilístico do algoritmo está intimamente ligado com essa dinâmica. Este estudo abordará três diferentes estratégias e os resultados alcançados através do PCV para uma instância específica de estudo.

4 Experimentos Computacionais

A utilização do algoritmo *Simulated Annealing* está muito condicionada na estratégia de variação das probabilidades de aceitação de soluções não melhores que a solução corrente. As probabilidades estão diretamente condicionadas ao esquema de queda de temperatura imposto ao algoritmo.

Usualmente o mecanismo de queda de temperatura é abordado através do termo do inglês *cooling schedule*. Este estudo tem interesse em verificar três diferentes estratégias de *cooling schedules* para utilização do algoritmo. Ao reduzir temperatura, a probabilidade de aceitação é modificada através da relação:

$$e^{-|F(\text{sol}_i) - F(\text{sol_new})|/T_i} \quad (4.1)$$

em que T_i é a temperatura na i -ésima iteração do algoritmo, $F(\bullet)$ é a função *fitness*, ou seja, função de avaliação das soluções, sol_i é a solução corrente e sol_new é a nova solução em avaliação.

Ao longo dos anos, vários métodos de redução de temperatura foram discutidos, entre os mais difundidos, é possível elencar, a redução geométrica, a redução exponencial, redução logarítmica, entre diversas outras. Na redução geométrica [27], o decaimento de temperatura é dependente de uma constante pré-fixada α , e a relação de queda de temperatura ao longo das iterações é dada por:

$$T_i = \alpha \times T_{i-1}, \text{ em que } 0 < \alpha < 1. \quad (4.2)$$

Já na redução baseada na adaptação do modelo de Lundy e Mees [28], o decaimento de temperatura é dependente de uma constante pré-fixada γ , e relação de queda de temperatura ao longo das iterações é dada por:

$$T_i = \frac{T_{i-1}}{1 + \gamma \sqrt{T_{i-1}}}, \text{ em que } 0 < \gamma < 1. \quad (4.3)$$

Por fim, na redução logarítmica, o decaimento de temperatura usado não era dependente de quaisquer constantes pré-fixadas. Cruz et al. (2018) [21] utilizam e apresentam uma descrição para este esquema de decaimento de temperatura, a a relação de queda de temperatura ao longo das iterações é dada por:

$$T_i = \frac{1}{\log(i+1)}. \quad (4.4)$$

Neste estudo, o resfriamento definido pela Equação 4.2 será denominado resfriamento 1, o resfriamento definido pela Equação 4.3 será denominado resfriamento 2 e o resfriamento definido pela Equação 4.4 será denominado resfriamento 3. A Figura 3 utiliza valores de α e γ fixados em 0,995 e 0,03, respectivamente. O interesse é estabelecer uma comparação visual da estrutura de decaimento de temperatura.

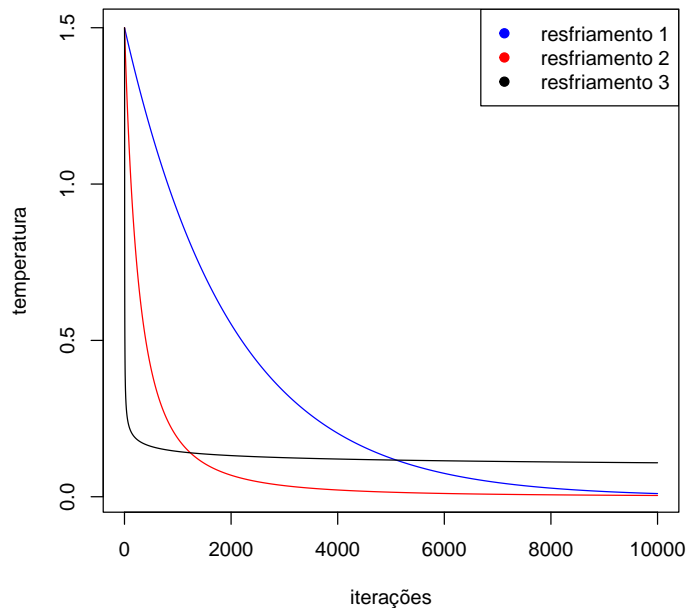


Figura 3 – Diferentes Estratégias de *Cooling Schedules*.

Dentre as três estratégias de resfriamento apresentadas, é possível verificar que o resfriamento 1 apresenta um decaimento mais lento de temperatura. O resfriamento 2 apresenta um decaimento mais acelerado. O resfriamento 3 tem uma propriedade específica, ele decai rapidamente nas primeiras iterações porém se aproxima da probabilidade zero muito mais lentamente que as demais estratégias.

Os algoritmos de otimização foram implementados na linguagem R, com a utilização do ambiente de desenvolvimento RStudio versão 4.3.2. Os códigos aqui utilizados foram gentilmente cedidos pelo professor Anderson Duarte, e estão disponíveis a pedido diretamente, para fins educacionais e de pesquisa. Os experimentos computacionais foram executados em um computador pessoal, processador 11th Gen Intel(R) Core(TM) i7-1165G7 @ 2.80GHz, 2 Núcleos, 4 Processadores Lógicos com sistema operacional Windows 11 Home Single 64 bits, com 8,00GB de memória RAM.

4.1 Calibração das Estratégias de Resfriamento 1 e 2

A tarefa inicial é a busca por uma escolha otimizada dos valores dos parâmetros α e γ nas estratégias de resfriamento 1 e 2. Um experimento foi conduzido baseado no

PCV descrito anteriormente. Uma instância específica foi produzida com o conjunto de bairros do município de Belo Horizonte no estado de Minas Gerais completamente geo-referenciados. Para cada bairro, um ponto fixo foi estabelecido e suas coordenadas são conhecidas. Dessa forma, é possível construir uma matriz de distâncias entre todos os bairros. Detalhamentos específicos sobre essa instância específica não são interesse central desse estudo. É importante fixar que trata-se somente de uma instância teste, sem quaisquer objetivos além da avaliação do algoritmo em uso. A Figura 4 apresenta resultados da calibração para o parâmetro α na estratégia de resfriamento 1.

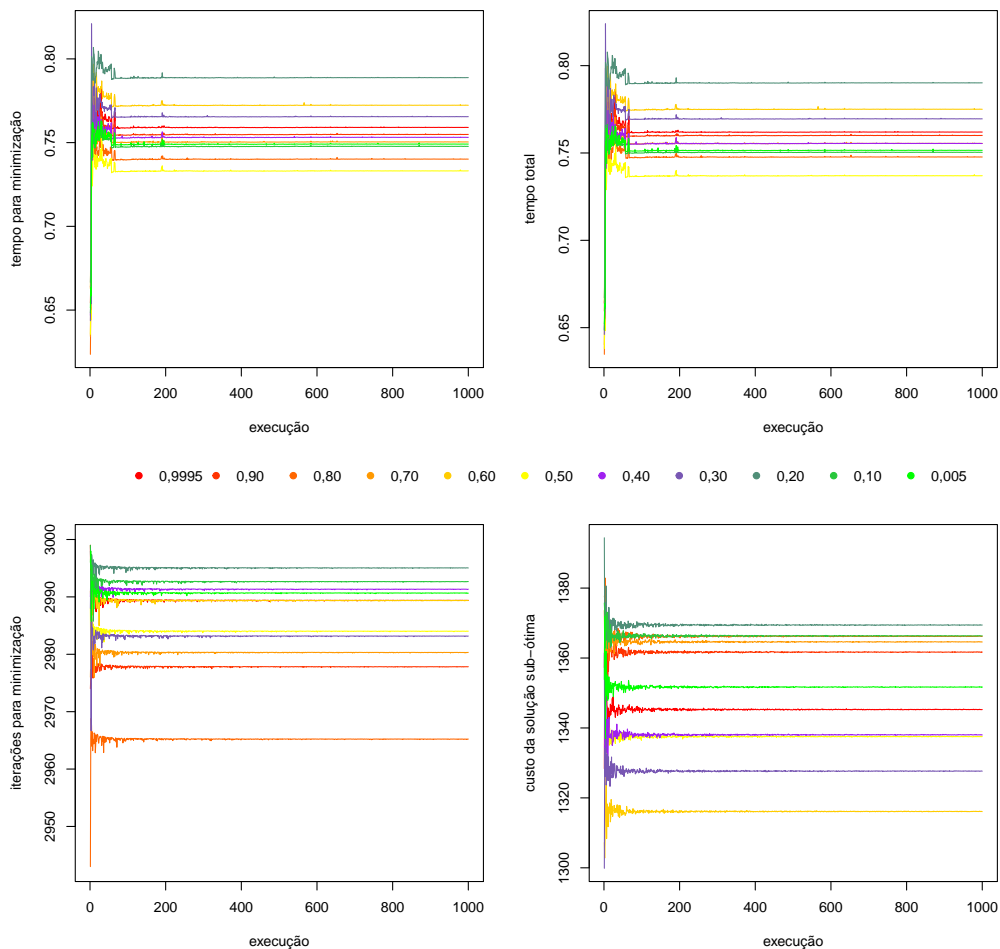


Figura 4 – Diferentes Valores de α para a Estratégia de Resfriamento 1.

Nessa etapa de calibração o procedimento de otimização foi realizado 1000 vezes e em cada uma dessas realizações, o algoritmo *Simulated Annealing* é executado com critério de parada em três mil execuções. O gráfico superior à esquerda apresenta o tempo necessário em segundos para o algoritmo obter a solução sub-ótima. Os valores apresentados são médias acumuladas entre as 1000 realizações. O gráfico superior à direita mostra o tempo médio total de execução, também em médias acumuladas. Além disso, o gráfico inferior à esquerda relata o número de iterações necessárias para alcançar a solução sub-ótima, também em médias acumuladas. Finalmente, o gráfico inferior à

direita apresenta o custo da solução sub-ótima obtida, também em médias acumuladas ao longo das 1000 realizações. Neste contexto, o custo aqui se refere à quilometragem do percurso solução para o PCV fornecido pelo algoritmo *Simulated Annealing*, essa mesma definição de custo é válida para as análises posteriores.

A similaridade entre os dois gráficos superiores mostra que a solução sub-ótima foi obtida na maior parte das execuções nas últimas iterações do algoritmo. Quando $\alpha = 0,99$ o tempo por execução é significativamente maior. Este tempo é quase 30% superior ao segundo maior tempo de execução verificado ($\alpha = 0,60$). A análise de quantidade de iterações não revela diferenças expressivas.

É possível verificar diferenças significativas para cada um dos valores do parâmetro α que foram utilizados quanto ao custo da solução sub-ótima fornecida. Para $\alpha = 0,50$ o custo da solução sub-ótima fornecida foi o menor. Além disso, para este valor de α os tempo de execução são também menores. O valor $\alpha = 0,99$ também forneceu custo baixo, porém com tempos de execução bem mais elevados. Após estas constatações, uma segunda calibração, mais minuciosa foi executada, foram testados valores na vizinhança de $\alpha = 0,50$. Estes resultados estão apresentados na Figura 5.

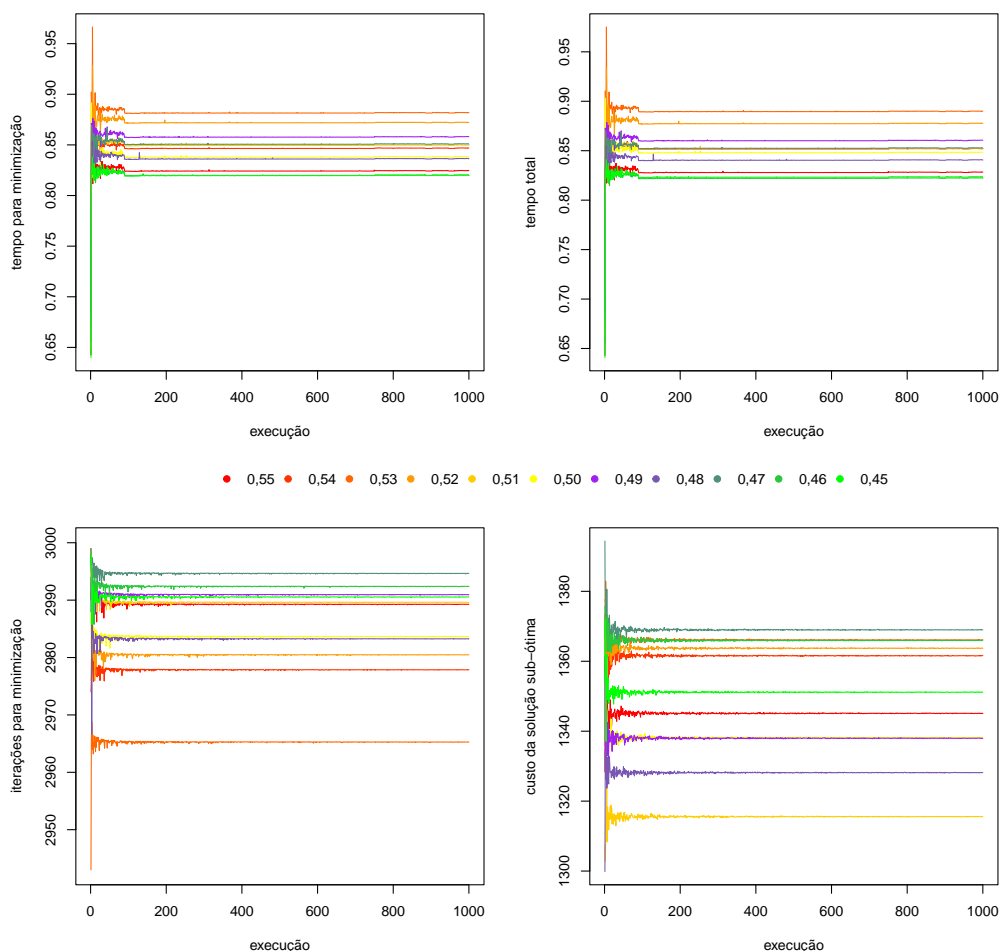


Figura 5 – Valores Vizinhos ao 0,50 para α com a Estratégia de Resfriamento 1.

O experimento com valores α vizinhos ($\alpha \in \{0,55; 0,54; 0,53; 0,52; 0,51; 0,50; 0,49; 0,48; 0,47; 0,46; 0,45\}$) ao valor mais adequado obtido na Figura 4 mostra que os menores valores de α apresentam resultados mais eficientes com respeito ao tempo de execução. O número de iterações para alcançar o máximo ficou similar para a maioria dos valores α . Notoriamente no gráfico para análise do custo médio das soluções obtidas, a parametrização com $\alpha = 0,50$ é superior. Em sequência, a estratégia 2 de resfriamento (ver Equação (4.3)) foi avaliada. A Figura 6 apresenta os resultados decorrentes da calibração para o parâmetro γ .

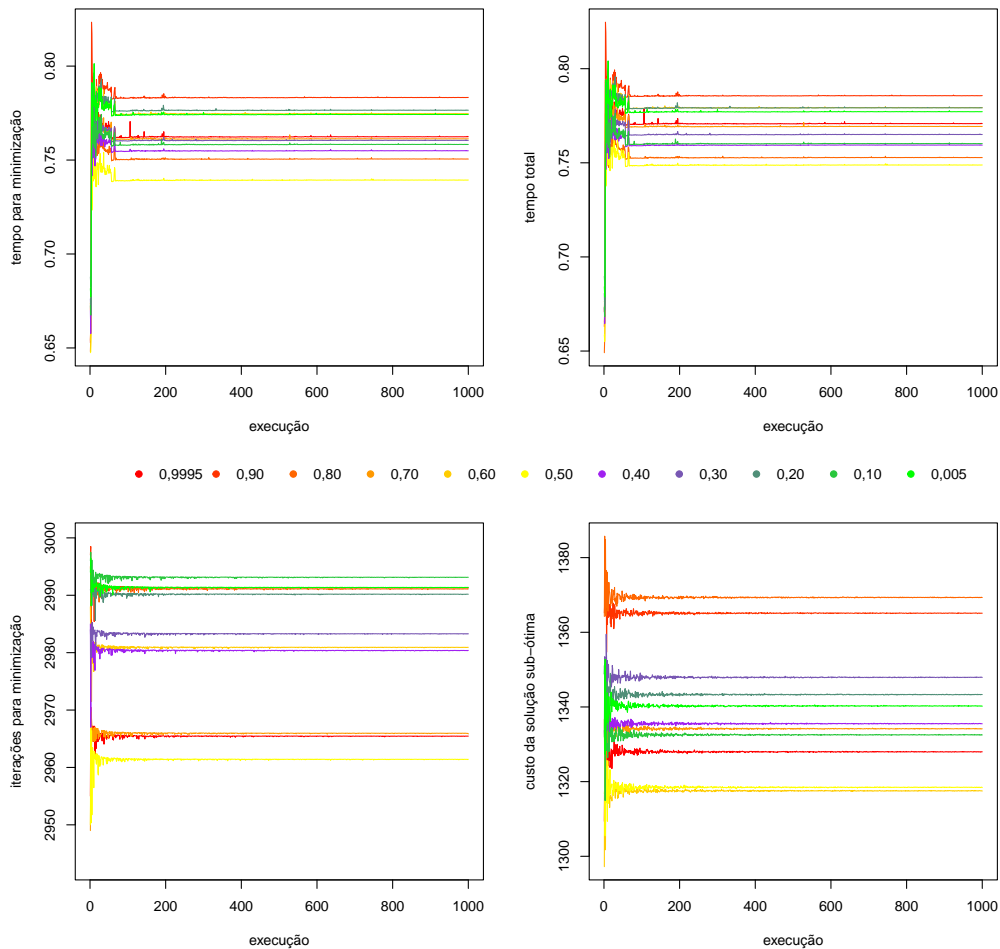


Figura 6 – Diferentes Valores de γ para a Estratégia de Resfriamento 2.

Na etapa de calibração de γ , novamente o procedimento de otimização foi realizado 1000 vezes e em cada uma dessas realizações, o algoritmo *Simulated Annealing* é executado com critério de parada em três mil execuções. Os quatro gráficos representam a mesma análise anterior. Novamente os gráficos de tempo de execução são bastante similares. Além disso, o número de iterações necessárias para obter a melhor solução ficou bem próximo do número máximo de iterações. Com respeito ao custo da melhor solução obtida, o valor $\gamma = 0,50$ e notoriamente mais destacado. Aqui também, após as constatações, uma segunda calibração, mais minuciosa foi executada, foram testados

valores na vizinhança de $\gamma = 0,50$. Estes resultados estão apresentados com detalhe na Figura 7.

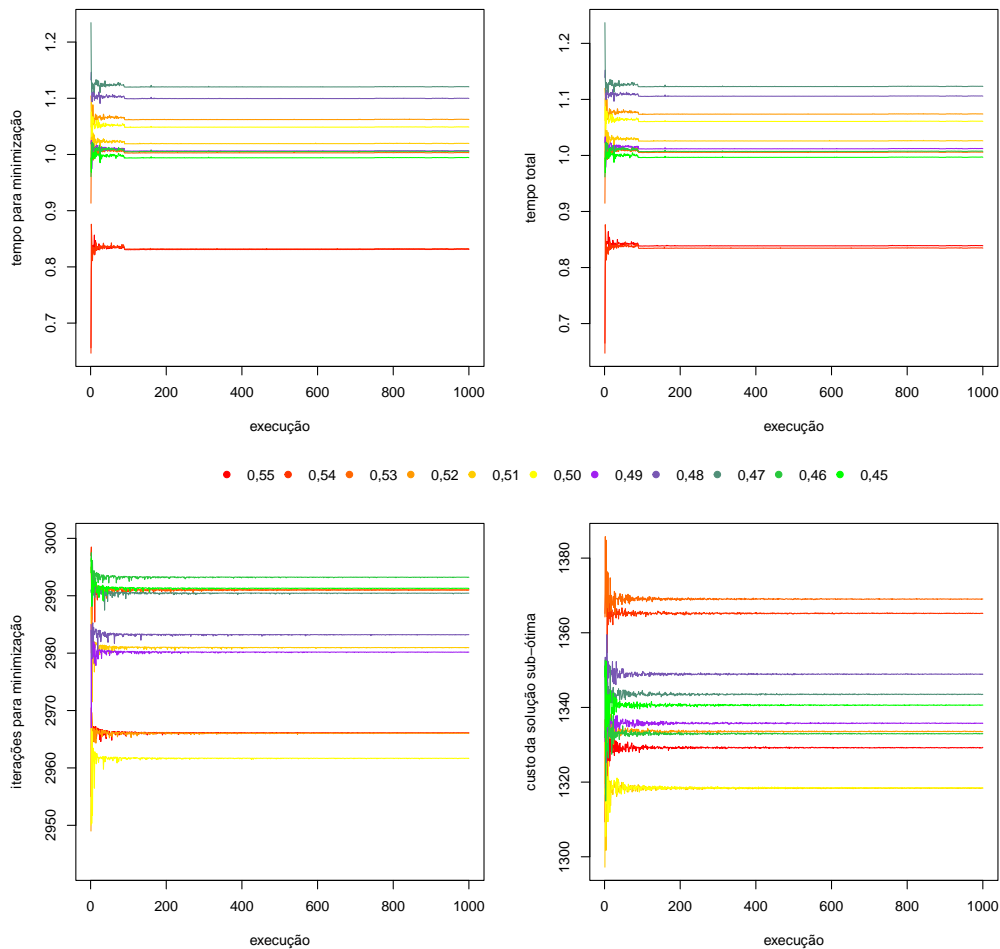


Figura 7 – Valores Vizinhos ao 0,50 para γ com a Estratégia de Resfriamento 2.

O experimento com valores γ vizinhos ($\gamma \in \{0,55; 0,54; 0,53; 0,52; 0,51; 0,50; 0,49; 0,48; 0,47; 0,46; 0,45\}$) ao valor mais adequado obtido na Figura 6 mostra que os maiores valores de γ apresentam resultados mais eficientes com respeito ao tempo de execução. O número de iterações para alcançar o máximo novamente ficou semelhante para todos os valores γ . Quanto ao gráfico para análise do custo médio das soluções obtidas, a parametrização com $\gamma = 0,50$ é superior, e com tempos de processamento abaixo significativamente menores que os verificados para os demais valores.

4.2 Comparação entre as Três Estratégias de Resfriamento

De posse de uma calibração efetiva de parâmetros, as equações de redução de temperatura dos esquemas 1 e 2 de resfriamento podem agora ser adotadas como

$$T_i = 0,50 \times T_{i-1}, \quad (4.5)$$

e também,

$$T_i = \frac{T_{i-1}}{1 + 0,50 \sqrt{T_{i-1}}}, \quad (4.6)$$

O esquema de redução logarítmica para o decaimento de temperatura não demanda parâmetros externos. Em outras palavras, a Equação (4.4) já determina completamente este esquema de resfriamento. Neste ponto, o interesse é comparar os resultados obtidos agora entre os três esquemas de redução de temperatura. A Figura 8 apresenta estes resultados.

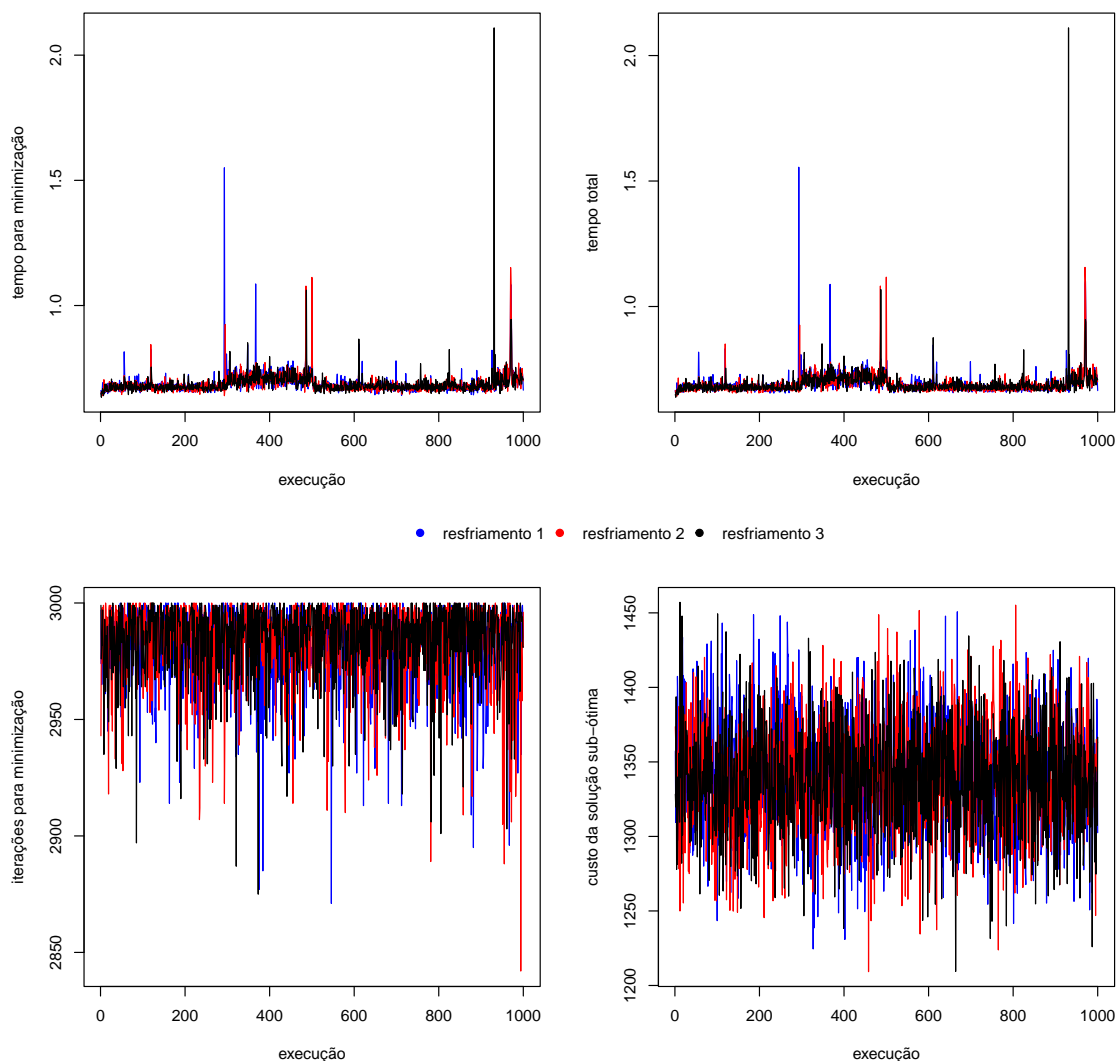


Figura 8 – Comparação de Resultados para os Esquemas de Redução de Temperatura 1, 2 e 3.

Ao analisar os resultados iniciais das simulações, é difícil constatar superioridade de algum método em relação aos demais. O elevado efeito de variabilidade entre as execuções individuais prejudica uma análise definitiva. Essa variabilidade é inerente ao procedimento de otimização. Como executado nos experimentos de calibração, o uso de

médias acumuladas atenua este efeito e possibilita uma análise clara da capacidade das metodologias. A Figura 9 apresenta resultados por média acumuladas com a comparação das três estratégias distintas de resfriamento e a solução inicial sem qualquer estratégia de otimização. A comparação é feita em termos do custo a solução apresentada.

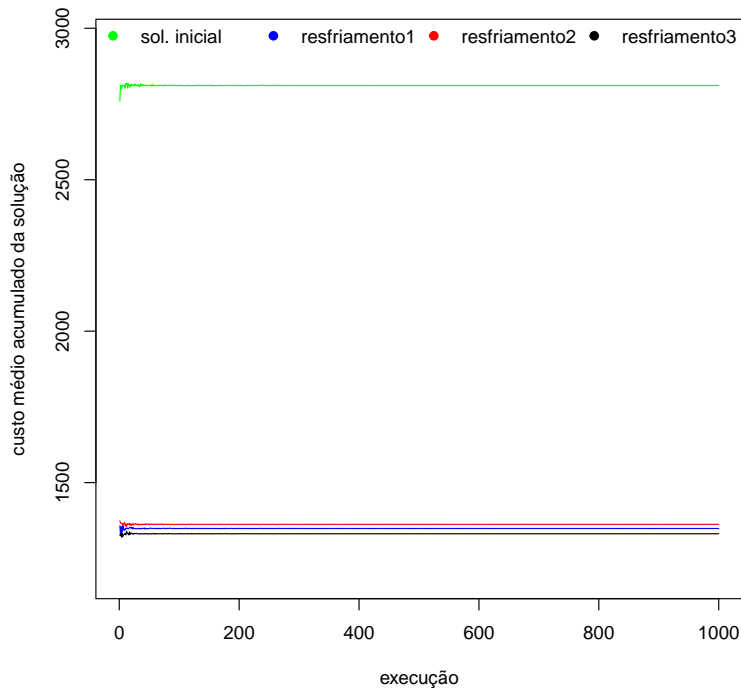


Figura 9 – Comparação de Custo Final das Soluções Fornecidas para os Esquemas de Redução de Temperatura 1, 2 e 3 e a Solução Inicial.

É possível verificar que independentemente da estratégia de resfriamento, o algoritmo *Simulated Annealing* é capaz de reduzir sobremaneira o custo de uma solução proposta sem qualquer estratégia de otimização. O custo médio das soluções otimizadas é aproximadamente 50% do custo das soluções prévias. Isso confirma que inicialmente que o algoritmo se adapta bem ao PCV, ou seja, configura uma estratégia de boa qualidade para fornecer soluções para este clássico problema da literatura.

Em seguida, representações gráficas similares às da Figura 8 são apresentadas porém com o uso de médias acumuladas. Ao optar por apresentar as médias acumuladas, o impacto da variabilidade é atenuado. Essa estratégia proporciona uma visão mais estável e clara do desempenho médio ao longo das execuções. Essa abordagem de agregação de dados permite que os resultados sejam interpretados com mais foco nos padrões gerais. Além disso, a média acumulada simplifica a comparação entre diferentes estratégias de resfriamento, e evidencia as diferenças de desempenho ao longo do tempo. Ao utilizar médias acumuladas, a intenção é fornecer uma representação mais compreensível e informativa dos resultados, que destaque os aspectos mais relevantes do desempenho das estratégias de resfriamento dentro do estudo. Os resultados podem

ser visualizados na Figura 10.

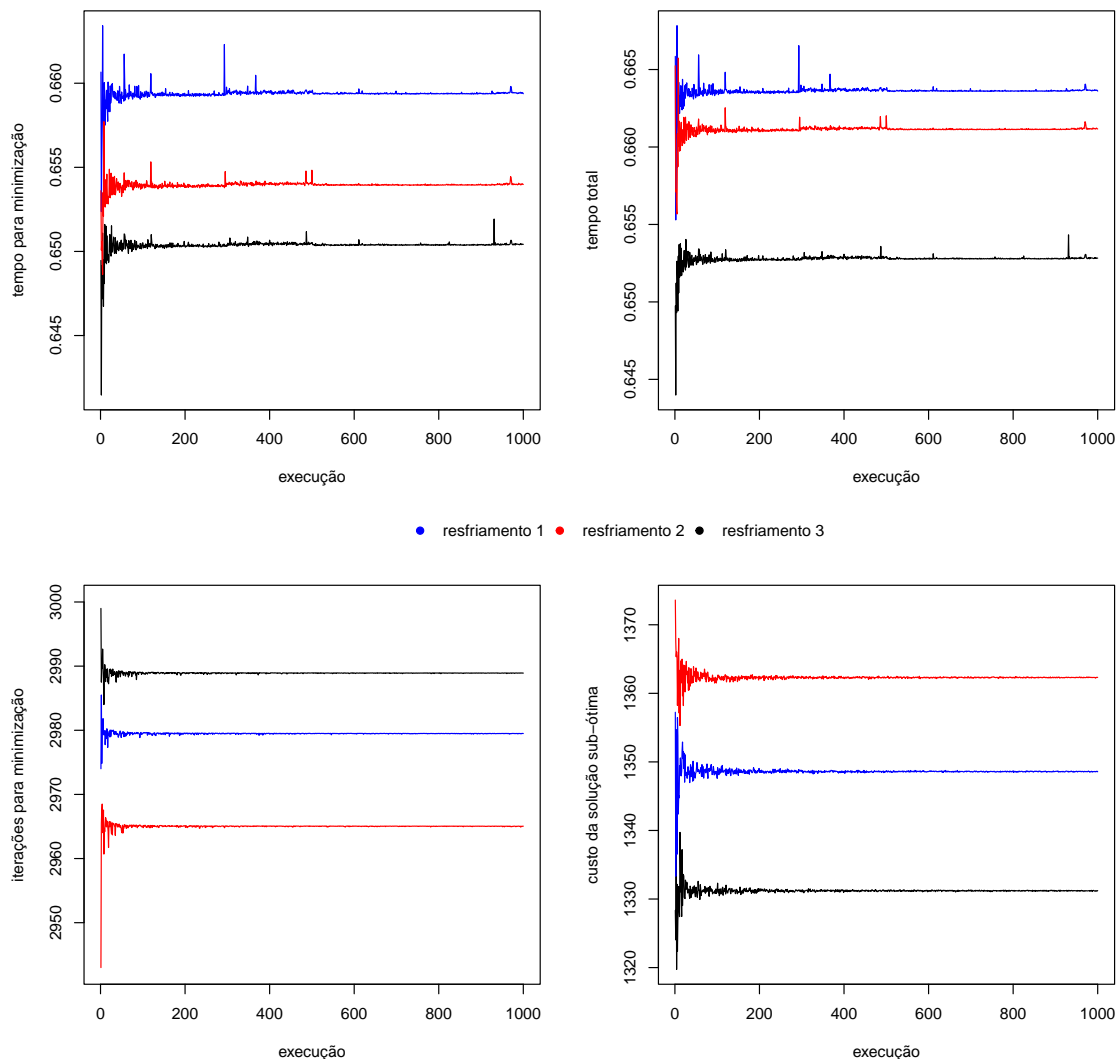


Figura 10 – Comparação de Resultados para os Esquemas de Redução de Temperatura 1, 2 e 3 por Médias Acumuladas.

Ao analisar os quatro gráficos da Figura 10, observa-se que o esquema de resfriamento 3 se destacou como o mais eficaz. Ele exibiu os menores tempos médios acumulados para minimização e tempo total acumulado. O número médio de iterações para minimização foi superior, mas sem acarretar em perda no tempo de processamento. Por fim, quanto ao custo médio acumulado da solução sub-ótima, o resfriamento 3 obteve os resultados superiores. Esses resultados indicam que a abordagem logarítmica de resfriamento, caracterizada por uma redução mais rápida nas primeiras iterações seguida por uma exploração mais completa ao longo do tempo pode ser mais eficiente em comparação com os esquemas de resfriamento geométrico e Lundy & Mees.

Nos gráficos de evolução temporal é possível verificar similaridade na forma, apesar de diferença no tempo médio. Este fato decorre da mesma solução inicial ser utilizada pelos três algoritmos. É fato que uma solução inicial de qualidade mais precária

afeta o desempenho do algoritmo e possa exigir um número maior de execuções. Por outro lado, este efeito pode também indicar a possibilidade de soluções iniciais mais precárias exigirem alguma estratégia algorítmica mais robusta que o *Simulated Annealing*.

Em contraste, o esquema geométrico, que presume uma taxa constante de resfriamento, e o esquema de Lundy & Mees, que busca um equilíbrio entre exploração global e local, apresentaram resultados também sólidos. Embora não superarem a eficácia do esquema logarítmico são também extremamente competitivos. O resfriamento geométrico, por sua natureza, pode ser mais adequado para cenários em que a convergência rápida é crucial, enquanto o Lundy & Mees pode ser preferível em situações que exigem uma exploração mais balanceada do espaço de busca.

5 Considerações Finais

O estudo e a implementação de uma aplicação computacional para otimização é um assunto de destacado interesse em estudos de Estatística, Ciência de Dados entre outras possíveis áreas de pesquisa. Além disso, esse tipo de desenvolvimento apresenta um nível de exigência de conhecimento de conceitos computacionais (técnicas algorítmicas e de estruturação de dados) e também de conhecimentos estatísticos (probabilidade geral) não completamente explorado durante um curso usual de graduação em Estatística.

Este estudo permitiu conhecer o bastante difundido algoritmo *Simulated Annealing* bem como algumas estratégias precursoras como o algoritmo *Metropolis*. Além disso, a investigação proporcionou uma análise aprofundada para algumas estratégias de resfriamento aplicadas ao clássico algoritmo *Simulated Annealing*.

Como base de estudo e teste, o conhecido Problema do Caixeiro Viajante (PCV) precisou ser abordado. O estudo elucidou que heurísticas, em particular o *Simulated Annealing*, desempenham um papel crucial na resolução eficiente de problemas de otimização combinatória complexos, como o PCV.

Durante a investigação das estratégias de resfriamento, três abordagens foram minuciosamente avaliadas: o resfriamento geométrico, o resfriamento de Lundy & Mees, e a estratégia logarítmica. A última destacou-se como a mais eficaz nesse estudo, forneceu soluções mais satisfatórias do que as soluções obtidas por meio dos outros dois mecanismos de resfriamentos. Sua capacidade de realizar uma rápida exploração inicial, seguida por uma exploração mais detalhada, evidenciou sua adaptabilidade e eficiência diante do PCV, e reforçou a importância de estratégias de resfriamento alinhadas à dinâmica do problema.

A sensibilidade dos resultados aos parâmetros das estratégias de resfriamento, notadamente α e γ , foi observada. Valores ótimos foram encontrados por meio de calibrações detalhadas. Essa escolha criteriosa revelou-se relevante para otimizar o desempenho do *Simulated Annealing*, o que destacou a importância do ajuste preciso desses parâmetros.

Finalmente, este estudo não apenas contribuiu para o avanço do conhecimento sobre a otimização de uma forma geral, mas particularmente em problemas combinatórios como o caso do PCV. Além disso também destacou a importância de escolhas estratégicas e ajustes precisos para maximizar a eficácia de heurísticas, como o *Simulated Annealing*, em problemas complexos de otimização combinatória.

5.1 Propostas para Trabalhos Futuros

Dentre possíveis propostas de continuidade de estudos, é possível enumerar:

- i) **comparação com outras heurísticas:** investigar o desempenho relativo do *Simulated Annealing* em comparação com outras heurísticas amplamente utilizadas;
- ii) **otimização multi-objetivo:** explorar abordagens de otimização multi-objetivo para o PCV, que considerem não apenas o custo da rota, mas também outros critérios relevantes, como tempo de execução ou número de veículos envolvidos no trajeto, abordagens que podem proporcionar soluções mais abrangentes e adaptáveis às necessidades específicas de diferentes aplicações;
- iii) **aplicações específicas:** investigar como as heurísticas, com a inclusão do *Simulated Annealing*, se comportam em cenários específicos relacionados ao PCV, como entregas em ambientes urbanos ou logística de transporte de mercadorias, personalizar as estratégias heurísticas para atender a requisitos específicos pode resultar em soluções mais eficientes e práticas.

Essas propostas buscam aprimorar a compreensão do desempenho do *Simulated Annealing* no contexto do PCV, ao mesmo tempo em que exploram alternativas e ampliam o escopo da pesquisa para aplicações práticas mais desafiadoras.

Referências

- [1] Kirkpatrick, Scott, C Daniel Gelatt Jr e Mario P Vecchi: *Optimization by simulated annealing*. *Science*, 220(4598):671–680, 1983. Citado na página 5.
- [2] Černý, Vladimír: *Thermodynamical approach to the traveling salesman problem: An efficient simulation algorithm*. *Journal of Optimization Theory and Applications*, 45(1):41–51, 1985. Citado na página 5.
- [3] Rahimunnisa, K: *Hybridized genetic-simulated annealing algorithm for performance optimization in wireless adhoc network*. *Journal of Soft Computing Paradigm (JSCP)*, 1(01):1–13, 2019. Citado na página 6.
- [4] Saffaran, Ali, Masoud Azadi Moghaddam e Farhad Kolahan: *Optimization of backpropagation neural network-based models in EDM process using particle swarm optimization and simulated annealing algorithms*. *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, 42:1–14, 2020. Citado na página 6.
- [5] He, Feng e Qing Ye: *A bearing fault diagnosis method based on wavelet packet transform and convolutional neural network optimized by simulated annealing algorithm*. *Sensors*, 22(4):1410, 2022. Citado na página 6.
- [6] Ji, Qingran, Fanchao Meng, Hongzhen Zheng e Dianhui Chu: *Optimization and Integration of Logistics Facilities Resources Based on Genetic-Simulated Annealing Hybrid Algorithm*. Em *2021 4th International Conference on Artificial Intelligence and Big Data (ICAIBD)*, páginas 135–142. IEEE, 2021. Citado na página 6.
- [7] Oudani, Mustapha: *A simulated annealing algorithm for intermodal transportation on incomplete networks*. *Applied Sciences*, 11(10):4467, 2021. Citado na página 6.
- [8] Yang, Li: *Research on logistics distribution vehicle path optimization based on simulated annealing algorithm*. *Advances in Multimedia*, 2022:1–8, 2022. Citado na página 6.
- [9] Manaswini, S e Gerard Deepak: *Towards a novel strategic scheme for web crawler design using simulated annealing and semantic techniques*. Em *Data Science and Security: Proceedings of IDSCS 2021*, páginas 468–477. Springer, 2021. Citado na página 6.
- [10] Haznedar, Bulent, Mustafa Turan Arslan e Adem Kalinli: *Optimizing ANFIS using simulated annealing algorithm for classification of microarray gene expression cancer data*. *Medical & Biological Engineering & Computing*, 59:497–509, 2021. Citado na página 6.

- [11] Chen, Chen e Lee Kong Tiong: *Using queuing theory and simulated annealing to design the facility layout in an AGV-based modular manufacturing system*. International Journal of Production Research, 57(17):5538–5555, 2019. Citado na página 6.
- [12] Toncovich, A, D Rossit e Mariano Frutos: *Solving a multi-objective manufacturing cell scheduling problem with the consideration of warehouses using a simulated annealing based procedure*. International Journal of Industrial Engineering Computations, 10(1):1–16, 2019. Citado na página 6.
- [13] Zhang, Yan, Xiaoxia Han, Yingchao Dong, Jun Xie, Gang Xie e Xinying Xu: *A novel state transition simulated annealing algorithm for the multiple traveling salesmen problem*. The Journal of Supercomputing, 77:11827–11852, 2021. Citado na página 6.
- [14] İlhan, İlhan e Gazi Gökmen: *A list-based simulated annealing algorithm with crossover operator for the traveling salesman problem*. Neural Computing and Applications, páginas 1–26, 2022. Citado na página 6.
- [15] Lee, Sangmin e Seoung Bum Kim: *Parallel simulated annealing with a greedy algorithm for Bayesian network structure learning*. IEEE Transactions on Knowledge and Data Engineering, 32(6):1157–1166, 2019. Citado na página 6.
- [16] Hosseini, Farzaneh Sajedi, Bahram Choubin, Amir Mosavi, Narjes Nabipour, Shahaboddin Shamshirband, Hamid Darabi e Ali Torabi Haghighi: *Flash-flood hazard assessment using ensembles and Bayesian-based machine learning models: Application of the simulated annealing feature selection method*. Science of the total environment, 711:135161, 2020. Citado na página 6.
- [17] Shen, Yadi, Yingchao Dong, Xiaoxia Han, Jinde Wu, Kun Xue, Meizhu Jin, Gang Xie e Xinying Xu: *Prediction model for methanation reaction conditions based on a state transition simulated annealing algorithm optimized extreme learning machine*. International Journal of Hydrogen Energy, 48(64):24560–24573, 2023. Citado na página 6.
- [18] Attiya, Ibrahim, Mohamed Abd Elaziz, Shengwu Xiong et al.: *Job scheduling in cloud computing using a modified harris hawks optimization and simulated annealing algorithm*. Computational intelligence and neuroscience, 2020, 2020. Citado na página 6.
- [19] Capraz, Seval, Halil Azyikmis e Adnan Ozsoy: *An optimized GPU-accelerated route planning of multi-UAV systems using simulated annealing*. International Journal of Machine Learning and Computing, 10(3), 2020. Citado na página 6.
- [20] Spinellis, D, C Papadopoulos e J M Smith: *Large production line optimization using simulated annealing*. International Journal of Production Research, 38(3):509–541, 2000. Citado na página 6.

- [21] Cruz, F R B, A R Duarte e G L Souza: *Multi-objective performance improvements of general finite single-server queueing networks*. *Journal of Heuristics*, 24(5):757–781, 2018. Citado 2 vezes nas páginas 6 and 19.
- [22] Amjath, Mohamed, Laoucine Kerbache, James MacGregor Smith e Adel Elomri: *Optimisation of Buffer Allocations in Manufacturing Systems: A Study on Intra and Outbound Logistics Systems Using Finite Queueing Networks*. *Applied Sciences*, 13(17):9525, 2023. Citado na página 6.
- [23] Bhattacharya, Rabi N e Edward C Waymire: *Stochastic processes with applications*. SIAM, 2009. Citado na página 9.
- [24] Metropolis, Nicholas, Arianna W Rosenbluth, Marshall N Rosenbluth, Augusta H Teller e Edward Teller: *Equation of state calculations by fast computing machines*. *The journal of chemical physics*, 21(6):1087–1092, 1953. Citado na página 12.
- [25] Cook, William J, David L Applegate, Robert E Bixby e Vasek Chvatal: *The traveling salesman problem: a computational study*. Princeton university press, 2011. Citado na página 16.
- [26] Hoffman, Karla L, Manfred Padberg e Giovanni Rinaldi: *Traveling salesman problem: The traveling salesman problem is to determine the route which will minimize the time (distance) of the trip*. Em Meyers, Robert A. (editor): *Encyclopedia of Physical Science and Technology (Third Edition)*, páginas 1573–1578. Academic Press, New York, third edition edição, 2003, ISBN 978-0-12-227410-7. <https://www.sciencedirect.com/topics/earth-and-planetary-sciences/traveling-salesman-problem>. Citado na página 16.
- [27] Bonomi, Ernesto e Jean Luc Lutton: *The N-city travelling salesman problem: Statistical mechanics and the Metropolis algorithm*. *SIAM review*, 26(4):551–568, 1984. Citado na página 19.
- [28] Lundy, Miranda e Alistair Mees: *Convergence of an annealing algorithm*. *Mathematical programming*, 34:111–124, 1986. Citado na página 19.