

# Metaheurísticas

## Simulated Annealing

Felipe Augusto Lima Reis

[felipe.reis@ifmg.edu.br](mailto:felipe.reis@ifmg.edu.br)



# Sumário

## 1 Simulated Annealing

# RECOZIMENTO SIMULADO (SIMULATED ANNEALING)

# Simulated Annealing

- Fundamentado em uma analogia com a termodinâmica;
- Recozimento (*annealing*), em metalurgia, corresponde ao processo de “aquecer um metal até o ponto de fusão e então resfriá-lo lentamente, permitindo que as moléculas alcancem uma configuração de baixa energia e formem uma estrutura cristalina, livre de defeitos” [Luzia and Rodrigues, 2009] [Luke, 2013].

# Simulated Annealing

- Foi desenvolvido em 1983 por Kirkpatrick et al. [Souza, 2011] [Kirkpatrick et al., 1983];
- Pode ser considerado uma derivação do algoritmo Metropolis
  - Este foi desenvolvido por Nicholas Metropolis et al., ex-participantes do projeto Manhattan [Luke, 2013];
  - Corresponde a um método de Cadeia de Markov Monte Carlo<sup>12</sup> (MCMC) para obter amostras aleatórias a partir de uma distribuição de probabilidade cuja amostragem direta é difícil;
  - A sequência pode ser usada para aproximar a distribuição<sup>3</sup> ou para calcular uma integral [Dietze, 2012].

---

<sup>1</sup>Tradução de *Markov chain Monte Carlo*, compreende um conjunto de algoritmos para amostragem a partir de uma distribuição de probabilidades.

<sup>2</sup>Distribuições de probabilidade descrevem o comportamento aleatórios de fenômenos dependentes do acaso.

<sup>3</sup>Entre as técnicas para aproximação da distribuição está a construção de histogramas.

# Simulated Annealing

- O algoritmo é considerado uma estratégia de busca local [Luzia and Rodrigues, 2009];
- A cada iteração, são gerados valores para duas soluções, a atual e uma escolhida, que são comparadas;
  - Soluções melhores que a atual são sempre aceitas;
  - Uma fração de soluções piores que a atual são aceitas, permitindo que o algoritmo escape de máximos locais [Luzia and Rodrigues, 2009] [Luke, 2013].

# Simulated Annealing

- Para um problema de minimização, a variação de valor da função objetivo ao mover-se de uma solução  $S$  para uma solução vizinha candidata  $S'$  é definida como:

$$\Delta S = f(S') - f(S)$$

- Temos dois valores possíveis para  $\Delta S$ 
  - $\Delta < 0$ : método aceita o movimento e a solução vizinha passa a ser a nova solução corrente
  - $\Delta \geq 0$ : a solução vizinha candidata poderá ser aceita com uma probabilidade  $e^{-\Delta/t}$ , onde  $t$  é um parâmetro denominado temperatura [Souza, 2011].

# Simulated Annealing

- A fórmula de distribuição do Simulated Annealing é baseada na Distribuição de Boltzmann<sup>4</sup>;
- Essa distribuição indica a probabilidade de sistema estar em um determinado estado em função da energia do estado e da temperatura do sistema [Frausto-Solis et al., 2016].

$$P(S_{novo}) = e^{(\Delta S/t)}$$

onde

- $S$ : solução atual, de qualidade superior;
- $S_{novo}$ : próxima solução do sistema;
- $t$ : temperatura de recozimento atual do sistema;

---

<sup>4</sup>Também conhecida como Distribuição de Gibbs.



# Simulated Annealing

- A probabilidade de aceitar uma nova solução pode ser dada por: [Luke, 2013] e [Luzia and Rodrigues, 2009]

$$P(t, R, S) = e^{\left(\frac{Qualidade(R) - Qualidade(S)}{t}\right)}$$

onde

- $t$ : parâmetro do algoritmo, correspondente à temperatura de recozimento, definida a cada iteração;
- $R$ : solução de qualidade inferior;
- $S$ : solução atual, de qualidade superior;
- $e$ : constante; número de Euler;

# Simulated Annealing

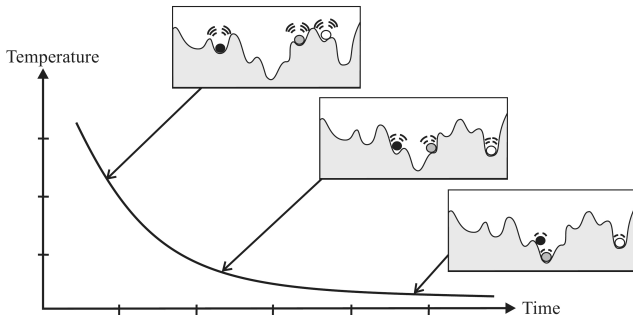
- A taxa de decrescimento (“temperatura”  $t$ ) é chamada de *schedule* do algoritmo e pode ser ajustado (*fine-tuning*);
- O parâmetro  $t$  inicia com um valor alto  $t_0$ , que vai gradualmente se reduzindo (para  $t \geq 0$ );
- A cada iteração, a temperatura é reduzida, o que diminui a probabilidade de escolha de uma solução menos promissora e aumenta a tendência de se melhorar a solução atual;
- Quando a temperatura atinge o valor zero<sup>5</sup>, o sistema é “congelado” e o mínimo de energia do sistema é atingido [Coppin, 2004] [Luke, 2013] [Luzia and Rodrigues, 2009].

---

<sup>5</sup>Na prática, em alguns sistemas, a temperatura não precisa atingir o valor zero. Quando a temperatura fica baixa o suficiente para não permitir a troca da solução, a execução do algoritmo pode ser finalizada [Souza, 2011].

# Simulated Annealing

- O possível comportamento e a influência da temperatura pode ser visto na imagem abaixo;
- Como o decrescimento nem sempre é “perfeito”, o aceite de soluções piores que a atual podem auxiliar na convergência.



Fonte: [da Silva, 2017]

# ESQUEMAS DE RESFRIAMENTO

# Esquemas de Resfriamento

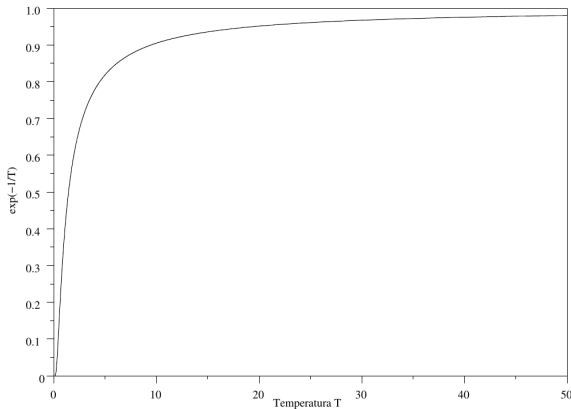
- Esquemas de Resfriamento<sup>6</sup> determinam taxa de decaimento da temperatura a cada etapa do algoritmo
  - Esses esquemas são essenciais na eficiência e efetividade do algoritmo [da Silva, 2017].
  - A taxa de resfriamento influencia a capacidade do algoritmo de fugir de mínimos (ou máximos) locais;
  - No entanto, o decrescimento lento fará com que o algoritmo leve inúmeras iterações até atingir o valor ótimo, tornando proibitiva a execução do algoritmo [Souza, 2011].

---

<sup>6</sup>Também denominados *cooling schedule* ou *cooling scheme*.

# Esquemas de Resfriamento

- A figura abaixo indica um possível comportamento de uma função de resfriamento.



Fonte: [Souza, 2011]

# Esquemas de Resfriamento

- Diferentes esquemas de resfriamento foram estudadas na literatura: [da Silva, 2017]
  - Linear:  $T_i = T_0 - i\beta$
  - Geométrico:  $T = \alpha T$
  - Logarítmico:  $T_i = \frac{T_0}{\log(i)}$
  - Decaimento lento:  $T_{i+1} = \frac{T_i}{1 + \beta T_i}$
- Outras funções de resfriamento podem ser encontradas na literatura<sup>7</sup>.

---

<sup>7</sup>Os trabalhos de [Souza, 2011] e [Peprah et al., 2017] contém diferentes funções de resfriamento.

# ALGORITMO



# Simulated Annealing

- O pseudo-algoritmo do Simulated Annealing pode ser visto abaixo;
- A possibilidade de aceitar soluções piores que a atual faz com que o sistema faça uma busca maior no espaço de soluções, fugindo de máximos e mínimos locais.

```
01:  $T \leftarrow$  temperatura com valor elevado
02:  $S \leftarrow$  solução candidata inicial qualquer
03: Melhor  $\leftarrow S$ 
04: repita
05:    $R \leftarrow$  GerarVizinho(  $S$  )
06:   se Qualidade( $R$ ) > Qualidade( $S$ ) ou se Aleatorio() <  $P(R,S,T)$  então
07:      $S \leftarrow R$ 
08:    $T \leftarrow$  NovaTemperatura( $T$ )
09:   se Qualidade( $S$ ) > Qualidade(Melhor) então
10:     Melhor  $\leftarrow S$ 
11: até que Melhor seja a solução ideal, ou o tempo tenha esgotado, ou  $T < 0$ 
12: devolva Melhor
```

*Simulated Annealing*

Fonte: [Luzia and Rodrigues, 2009]

## VANTAGENS E DESVANTAGENS

# Vantagens e Desvantagens

- Vantagens:
  - Se a temperatura  $t$  é reduzida de forma suficientemente lenta, então o sistema pode atingir o equilíbrio;
  - O algoritmo tem garantia de convergência (desde que a temperatura seja reduzida de forma suficientemente lenta) [Luzia and Rodrigues, 2009];
- Desvantagens:
  - Apesar da convergência ser garantida, ela pode ser muito lenta;
  - O “resfriamento rápido” pode não garantir a convergência da solução [Luzia and Rodrigues, 2009].

# Referências I



Coppin, B. (2004).

Artificial intelligence illuminated.

Jones and Bartlett illuminated series. Jones and Bartlett Publishers, 1 edition.



da Silva, D. M. (2017).

Métodos Heurísticos - Simulated Annealing - Slides de Aula.

IFMG - Instituto Federal de Minas Gerais, Campus Formiga.



Dietze, M. (2012).

Mcmc: Metropolis algorithm.

[Online]; acessado em 10 de Maio de 2021. Disponível em:

[https://www.life.illinois.edu/dietze/Lectures2012/Lesson12\\_Metropolis.pdf](https://www.life.illinois.edu/dietze/Lectures2012/Lesson12_Metropolis.pdf).



Frausto-Solis, J., Liñán-García, E., Sánchez-Hernández, J. P., González-Barbosa, J. J., González-Flores, C., and Castilla-Valdez, G. (2016).

Multiphase simulated annealing based on boltzmann and bose-einstein distribution applied to protein folding problem.

Advances in Bioinformatics, 2016.



Kirkpatrick, S., Gelatt, C., and Vecchi, M. (1983).

Optimization by simulated annealing.

In Fischler, M. A. and Firschein, O., editors, Readings in Computer Vision, pages 606–615. Morgan Kaufmann, San Francisco (CA).

## Referências II



Luke, S. (2013).

Essentials of Metaheuristics (Second Edition).

lulu.com, 2 edition.

[Online]; Disponível em: <https://cs.gmu.edu/~sean/book/metaheuristics/>.



Luzia, L. F. and Rodrigues, M. C. (2009).

Estudo sobre as metaheurísticas.

[Online]; acessado em 22 de Setembro de 2020. Disponível em:

<https://www.ime.usp.br/~gold/cursos/2009/mac5758/LeandroMauricioHeuristica.pdf>.



Peprah, A., Appiah, S., and Amponsah, S. (2017).

An optimal cooling schedule using a simulated annealing based approach.

Applied Mathematics, 8:1195–1210.



Souza, M. J. F. (2011).

Inteligência computacional para otimização.

[Online]; acessado em 12 de Maio de 2021. Disponível em: <http://www.decom.ufop.br/prof/marcone/Disciplinas/InteligenciaComputacional/InteligenciaComputacional.pdf>.