

# UTILIZAÇÃO DO ALGORITMO SIMULATED ANNEALING EM PROBLEMAS DE OTIMIZAÇÃO

Victor Ferreira da Silva<sup>1</sup>, Anderson Ribeiro Duarte<sup>1</sup>

**Resumo:** *Este trabalho tem um caráter bastante introdutório. Não sendo de interesse um estudo profundo a cerca dos itens que serão mencionados. Os objetivos principais giram em torno de apresentar os detalhes básicos da utilização do algoritmo Simulated Annealing e sugerir um problema real para sua utilização. Inicialmente é apresentada a idéia central que permeia a proposta do algoritmo. Posteriormente são citados alguns detalhes básicos sobre a sua formulação. Sua implementação é discutida na seção metodológica. Uma instância pequena do clássico problema do caixeiro viajante é mencionada a título de exemplo e o algoritmo é implementado para solucionar tal exemplo. É proposto um problema aplicado em Teoria das Filas para ser solucionado através do algoritmo proposto e Conclusões Finais são expostas.*

**Palavras-chave:** Simulated Annealing, Otimização, Teoria das Filas.

## Introdução

Para a apresentação do algoritmo *Simulated Annealing*, se faz necessária uma discussão inicial sobre tal nomenclatura. Pela definição, o nome recozimento (*annealing*) é dado ao processo de aquecimento de um sólido até o seu ponto de fusão, seguido de um resfriamento gradual e vagaroso, até que se alcance novamente o seu enrijecimento.

Computacionalmente falando o processo consiste em um procedimento estocástico (em um jargão simples o mesmo que aleatório) desta maneira procura-se encontrar um estado ideal que seja razoavelmente bom para se tomar como uma otimização de fato.

Simulated Annealing é considerado um tipo de algoritmo conhecido como de busca local. Ele se constitui em um método de obtenção de boas soluções para problemas de otimização vistos como problemas de difícil resolução. Desde a sua introdução como um método de otimização combinatorial, ele vem sendo vastamente utilizado em diversas áreas, tais como projeto de circuitos integrados auxiliado por computador, processamento de imagem, redes neurais, etc.

A sua semelhança com o método original no qual foi inspirado é muito grande. Na sua apresentação, nos trabalhos independentes de Kirkpatrick et al.(1983)[6] e Cerny (1985)[2], é mostrado como um modelo de simulação de recozimento de sólidos, como proposto em Metropolis et al.(1953)[7]. Pode ser utilizado em problemas de otimização, onde a função objetivo, a ser minimizada, corresponde à energia dos estados do sólido

Na utilização desse método, muito simples e rápido, um problema aparente é que o mínimo local encontrado pode estar longe de ser um mínimo global, o que se traduziria em uma solução inaceitável para o problema. Uma estratégia muito simples de aprimorar a solução obtida através desse tipo de algoritmo seria escolher a menor solução, de um conjunto de soluções obtidas de execuções sucessivas, realizadas a partir de diferentes soluções iniciais.

---

<sup>1</sup>Departamento de Matemática, ICEB, UFOP,  
victor020989@hotmail.com, anderson@iceb.ufop.br

O método Simulated Annealing não utiliza uma “estratégia” (uma lei por exemplo para convergência total), assumindo assim na maioria vezes um mínimo ou máximo que não é o global, mais se configura como uma boa opção para solução do problema em questão.

## Metodologia

Uma abordagem bastante interessante pode ser encontrada em Ross (2006)[8]. Utilizando o formato proposto na referência anterior, serão apresentados a seguir os conceitos básicos da modelagem do problema e a proposta do algoritmo Simulated Annealing.

### Modelagem via Simulated Annealing

Seja  $\mathcal{A}$  um conjunto finito de vetores e seja  $V(x)$  uma função não-negativa definida para  $x \in \mathcal{A}$ , suponha que estamos interessados em encontrar o seu valor máximo ou então o argumento  $x$  em que o valor máximo é atingido, ou seja,

$$V^* = \max_{x \in \mathcal{A}} V(x) \quad \text{e} \quad \mathcal{M} = \{x \in \mathcal{A} : V(x) = V^*\}$$

busca-se encontrar  $V^*$ , bem como um elemento em  $\mathcal{M}$ . Será mostrado agora como isso pode ser feito através de Simulated Annealing.

Inicialmente, seja  $\lambda > 0$  e considere a seguinte função de probabilidade no conjunto de valores em  $\mathcal{A}$ :

$$p_\lambda(x) = \frac{e^{\lambda V(x)}}{\sum_{x \in \mathcal{A}} e^{\lambda V(x)}}$$

Multiplicando o numerador e o denominador da forma anterior por  $e^{-\lambda V^*}$  e utilizando  $|\mathcal{M}|$  para denotar o número de elementos de  $\mathcal{M}$ , vemos que:

$$p_\lambda(x) = \frac{e^{\lambda(V(x)-V^*)}}{|\mathcal{M}| + \sum_{x \notin \mathcal{M}} e^{\lambda(V(x)-V^*)}}$$

entretanto, uma vez que  $V(x) - V^* < 0$  para  $x \notin \mathcal{M}$ , obtemos que,  $\lambda \rightarrow \infty$ ,

$$p_\lambda(x) \rightarrow \frac{\delta(x, \mathcal{M})}{|\mathcal{M}|} \quad \text{em que} \quad \delta(x, \mathcal{M}) = \begin{cases} 1 & \text{se } x \in \mathcal{M} \\ 0 & \text{se } x \notin \mathcal{M} \end{cases}$$

Assim, deixando o valor  $\lambda$  ser arbitrariamente grande e gerando uma cadeia de Markov cuja distribuição limite é  $p_\lambda(x)$ , então a maior parte da massa de probabilidade da distribuição limite se concentrará em pontos de  $\mathcal{M}$ . Uma abordagem que é frequentemente útil na definição de tal cadeia é introduzir o conceito de vetores vizinhos e, em seguida, usar o algoritmo de Metropolis-Hastings. Por exemplo, pode-se dizer que os dois vetores  $x \in \mathcal{A}$  e  $y \in \mathcal{A}$  são vizinhos se eles diferem em apenas uma única coordenada, ou se um pode ser obtido a partir do outro, trocando dois dos seus componentes. Nesta situação, o próximo estado a partir de  $x$  seria algum dos estados vizinhos de forma equiprovável. A escolha de um novo estado será segundo o seguinte procedimento, escolhe-se um vizinho  $y$ , então o próximo estado será  $y$  com a seguinte probabilidade:

$$\min \left\{ 1, \frac{e^{\lambda V(y)} / |N(y)|}{e^{\lambda V(x)} / |N(x)|} \right\}$$

caso contrário a cadeia permanece no estado  $x$ . Na definição anterior,  $|N(\cdot)|$  é o número de vizinhos de  $(\cdot)$ . Suponha que o estado  $y$ , seja escolhido aleatoriamente; se  $V(y) \geq V(x)$ , em

seguida, a cadeia se move para o estado  $y$ . Já se  $V(y) < V(x)$ , em seguida, a cadeia se move para o estado  $y$  estado com probabilidade  $e^{\lambda(V(y)-V(x))}$  ou permanece no estado  $x$  caso contrário.

Uma deficiência do algoritmo anterior é que, como foi escolhido um valor  $\lambda$  arbitrariamente grande, se a cadeia entra em um estado  $x$  cujo valor  $V(x)$  é maior do que  $V(\cdot)$  para cada um dos seus vizinhos, então pode demorar muito tempo para a cadeia passar para um estado diferente, ou seja, enquanto um grande valor de  $\lambda$  é necessário para a distribuição limite colocar mais da sua massa de probabilidade nos pontos em  $|\mathcal{M}|$ , esse valor normalmente requer um número muito grande de transições antes da distribuição limite ser atingida. Um segundo ponto fraco é que, desde que exista apenas um número finito de valores possíveis de  $x$ , todo o conceito de convergência parece sem sentido uma vez que pode sempre, em teoria, utilizar uma busca exaustiva dentre as possíveis soluções e assim obter a convergência em um número finito de etapas. Assim, ao invés de considerar a proposta anterior, de um ponto de vista estritamente matemático, faz mais sentido considerá-la como uma abordagem heurística, e encontrar um valor  $\lambda$  usual que se atualize com o tempo.

O Simulated Annealing é uma variação bastante difundida deste método e opera da seguinte forma:

Se o  $n$ -ésimo estado da cadeia de Markov é  $x$ , então algum vizinho de  $x$  é selecionado aleatoriamente. Se o estado vizinho escolhido for  $y$ , então o próximo estado da cadeia será  $y$  com probabilidade:

$$\min \left\{ 1, \frac{e^{\lambda_n V(y)/|N(y)|}}{e^{\lambda_n V(x)/|N(x)|}} \right\}$$

ou  $x$  caso contrário, nesta definição temos que  $\lambda_n$  para  $n \geq 1$  é fixado em um conjunto de valores que começam pequenos (resultando em um grande número de mudanças de estado) e depois crescem com as sucessivas transições.

Uma escolha computacionalmente usual de  $\lambda$  é  $C \log(1+n)$  em que  $C$  é alguma constante positiva pré-fixada (veja Besag et al.(1995)[1] e Diaconis and Holmes (1995)[4]).

Gerando  $m$  estados sucessivos  $Z_1, \dots, Z_m$ , pode-se então estimar  $V^*$  por:  $\max_{i=1, \dots, m} V(Z_i)$ , e se o máximo ocorre em  $Z_{i^*}$ , então este é tomado como um ponto estimado em  $\mathcal{M}$ .

## O Problema do Caixeiro Viajante

Uma versão do problema do caixeiro viajante é considerar um vendedor que inicia uma sequência de visitas à cidades iniciando na cidade 0 e, em seguida, sequencialmente visitando todas as cidades  $1, \dots, r$ . Uma escolha possível é então uma permutação  $X_1, \dots, X_r$  de  $1, \dots, r$  com a interpretação de que a partir de 0, o vendedor vai para a cidade  $X_1$ , em seguida,  $X_2$  e assim por diante. Supondo que a recompensa não negativa  $V(i, j)$  é o ganho sempre que o vendedor vai diretamente da cidade  $i$  à cidade  $j$ , então o retorno da possível sequência escolhida  $x = (X_1, \dots, X_r)$  é dado por:

$$V(x) = \sum_{i=1}^r v(X_{i-1}, X_i) \quad \text{em que } X_0 = 0$$

Duas permutações serão consideradas vizinhas se uma é resultado de um intercâmbio entre duas das coordenadas da outra, pode-se usar o Simulated Annealing para aproximar o melhor caminho, ou seja, começar com alguma permutação  $x$ , portanto  $Z_0 = x$ . Agora, uma vez que o  $n$ -ésimo estado foi obtido, então será gerado de forma aleatória algum vizinho do atual estado da seguinte forma:

Escolhe-se de forma aleatória os valores  $i$  e  $j$  no conjunto  $\{1, \dots, r\}$  considerando que os valores são equiprováveis e que  $i$  deve ser diferente de  $j$ . Posteriormente se constroi o vizinho de  $Z_n$  através da troca entre as coordenadas  $i$  e  $j$  de  $Z_n$ , este será o vizinho  $y$  de  $Z_n$ .

Agora é necessário determinar se a cadeia terá o  $(n + 1)$ -ésimo estado igual a  $y$ , para tanto escolhe-se de forma aleatória (uniforme) um valor  $u$  no intervalo  $[0, 1]$ , então:

$$Z_{n+1} = \begin{cases} y & \text{se } V(y) \geq V(Z_n) \\ y & \text{se } V(y) < V(Z_n) \text{ e } u \leq (1+n)^{V(y)-V(Z_n)} \\ Z_n & \text{se } V(y) < V(Z_n) \text{ e } u > (1+n)^{V(y)-V(Z_n)} \end{cases}$$

note que está sendo usado  $\lambda = \log(1 + n)$ .

## Um problema aplicado

Um problema importante para a aplicação do Simulated Annealing pode ser encontrado no estudo de Teoria das Filas. Suponha uma central de atendimento com espaço total para alocar  $K$  pessoas na fila de espera. Entretanto a central de atendimento deve ser subdividida em  $p$  estações de atendimento, cada uma destas estações faz um tipo distinto de atendimento, portanto têm taxas distintas de chegada de pessoas e taxas distintas para o tempo de atendimento. Considerando que cada estação tem a chegada de clientes ocorrendo segundo a taxa  $\lambda_i$  com  $i \in \{1, \dots, p\}$ , buscamos os valores  $k_i$ , tamanho da fila para cada uma das estações que preservem  $K = \sum_{i=1}^p k_i$  e também minimizem a soma das probabilidades de bloqueio (chance de chegada de um cliente no instante que a fila já está com todos os lugares ocupados) em cada uma das estações.

Cruz et al.(2008)[3] propõem uma aproximação eficiente para a probabilidade de bloqueio em uma fila  $M/G/1/k$ , da notação de Kendall (1953)[5],  $M$  entrada markoviana,  $G$  atendimento geral, 1 um único servidor e com espaço de tamanho  $k$ (unidades alocadas) na fila.

A aproximação proposta é dada por:

$$p_k = \frac{\rho \left( \frac{2 + \sqrt{\rho} c_s^2 - \sqrt{\rho} + 2(k-1)}{2 + \sqrt{\rho} c_s^2 - \sqrt{\rho}} \right) (-1 + \rho)}{\rho \left( \frac{2 + \sqrt{\rho} c_s^2 - \sqrt{\rho} + (k-1)}{2 + \sqrt{\rho} c_s^2 - \sqrt{\rho}} \right) - 1}$$

em que  $\rho = \frac{\lambda}{\mu}$  é a intensidade do tráfego na estação de atendimento (taxa de utilização da estação) para uma taxa de chegada  $\lambda$  e uma taxa de atendimento (serviço)  $\mu$  e ainda,  $c_s^2 = \frac{\text{Var}(T_s)}{\text{E}^2(T_s)}$  quadrado do coeficiente de variação do tempo de atendimento  $T_s$  utilizado para caracterizar sua distribuição geral.

O problema então será modelado da seguinte forma:

$$V^* = \min V(x)$$

em que:

$$x = (k_1, \dots, k_p);$$

$$K = \sum_{i=1}^p k_i \text{ com } k_i \in \{0, 1, \dots, K\};$$

$$V(x) = \sum_{i=1}^p p_{k_i} \text{ com } p_{k_i} = \frac{\rho_i \left( \frac{2 + \sqrt{\rho_i} c_{s_i}^2 - \sqrt{\rho_i} + 2(k_i-1)}{2 + \sqrt{\rho_i} c_{s_i}^2 - \sqrt{\rho_i}} \right) (-1 + \rho_i)}{\rho_i \left( \frac{2 + \sqrt{\rho_i} c_{s_i}^2 - \sqrt{\rho_i} + (k_i-1)}{2 + \sqrt{\rho_i} c_{s_i}^2 - \sqrt{\rho_i}} \right) - 1}$$

Note que modificações na alocação individual de espaço entre as filas, mantendo fixo o espaço total alocado levará a alterações no valor da função objetivo  $V$ .

Neste caso, se torna importante a definição de vizinhança para uma possível alocação  $(k_1, \dots, k_p)$ . Dado dois valores aleatórios  $i$  e  $j$  em  $\{1, \dots, p\}$ , um dos vizinhos da alocação anterior é dado por  $(k_1, \dots, k_{i-1}, k_i - 1, k_{i+1}, \dots, k_{j-1}, k_j + 1, k_{j+1}, \dots, k_p)$ , neste procedimento é importante notar que uma configuração somente é considerada factível se suas coordenadas estão em  $\{1, \dots, K\}$ .

Desta forma será bastante relevante propor uma busca dentre as possíveis alocações  $(k_1, \dots, k_p)$  através do algoritmo Simulated Annealing, visando obter a configuração que minimiza a soma das probabilidades de bloqueio pensando em um espaço alocado total pré-fixado  $K$ .

## Conclusões

As idéias aqui discutidas são frutos de um projeto de iniciação científica ainda embrionário, portanto a discussão ainda se encontra em estágio preliminar, entretanto, algumas conclusões já foram observadas.

O algoritmo Simulated Annealing é uma proposta de algoritmo estocástico e, como todas as outras propostas deste gênero, não garante obtenção de ótimos globais, mas apresenta boas soluções em tempo computacional satisfatório.

O problema do caixeiro viajante discutido anteriormente já foi amplamente discutido e didaticamente se torna um excelente exemplo para a discussão inicial do algoritmo Simulated Annealing. Uma versão para o problema com a instância aqui discutida já foi implementada e forneceu resultados bastante interessantes. Entretanto por se tratar de um exemplo puramente didático neste trabalho, os resultados obtidos não foram mencionados.

O problema sugerido para ser otimizado através da técnica Simulated Annealing é um tema de bastante relevância e acredita-se que as soluções obtidas serão de grande valia. Além disso utiliza-se de um artigo recente mostrando que são propostas bastante atualizadas.

Espera-se com este trabalho não apenas apresentar uma ilustração sobre o algoritmo Simulated Annealing, como também, propor discussão sobre problemas específicos de aplicação que podem ser solucionados através desta técnica.

Já temos a implementação em linguagem C++ da proposta Simulated Annealing para o problema do caixeiro viajante.

## Referências

- [1] BESAG, J., GREEN, P., HIGDON, D., AND MENGERSEN, K., Bayesian Computation and Stochastic Systems (with Discussion), *Statistical Sci.*, **10**, 3-67, 1995.
- [2] CERNY, V., Thermodynamical approach to the traveling salesman problem: an efficient simulation algorithm, *Journal of Optimization Theory and Applications*, **45**, 41-51, 1985.
- [3] CRUZ, F.R.B., DUARTE, A.R. AND VAN WOENSEL, T., Buffer allocation in general single-server queueing network, *Computer and Operations Research*, **35(11)**, 3581-3598, 2008.
- [4] DIACONIS, P. AND HOLMES, S., Three Examples of Monte-Carlo Markov Chains: At the Interface between Statistical Computing, Computer Science, and Statistical Mechanics, em "Discrete Probability and Algorithms" (Aldous, D., Diaconis, P., Spence, J. and Steele, J. M., eds.) pp. 43-56, Springer-Verlag, 1995.
- [5] KENDALL, D. G., Stochastic processes occurring in the theory of queues and their analysis by the method of imbedded markov chains, *Annals Mathematical Statistics*, **24**, 338-354, 1953.
- [6] KIRKPATRICK, S., GELATT, C., AND VECCHI, M., Optimization by simulated annealing, *Science*, **4598**, 671-680, 1983.

- [7] METROPOLIS, N., A.W. ROSENBLUTH, M.N. ROSENBLUTH, A.H. TELLER, E. TELLER, Equations of state calculations by fast computing machines, *Journal of Chemical Physics*, **21**, 1087-1091, 1953.
- [8] ROSS, S. Simulation, London: Elsevier Academic Press, Fourth Edition, 2006.

## **Agradecimentos**

Os autores agradecem o apoio recebido através do projeto de iniciação científica PIBIC/Al apoiado pelo CNPq.