

**Método de simulated annealing para
solução aproximada de equações diferenciais**

Melba Luz Torres Ortiz

DISSERTAÇÃO APRESENTADA
AO
INSTITUTO DE MATEMÁTICA E ESTATÍSTICA
DA
UNIVERSIDADE DE SÃO PAULO
PARA
OBTENÇÃO DO TÍTULO
DE
MESTRE EM CIÊNCIAS

Programa: Estatística

Orientador: Prof. Anatoli Iambartsev (IME-USP)

Coorientador: Prof. Eugene Pechersky ((Russian Academy of
Sciences, Moscow, Russia.))

Durante o desenvolvimento deste trabalho o autor recebeu auxílio financeiro da CNPq.

São Paulo, novembro de 2017

Agradecimentos

This work was supported by CNPq.

Resumo

Melba Luz Torres Ortiz **Método de simulated annealing para solução aproximada de equações diferenciais..** 2017. 19 f. Dissertação (Mestrado) - Instituto de Matemática e Estatística, Universidade de São Paulo, São Paulo, 2017.

Propomos um método numérico para resolver as equações diferenciais parciais, em particular, equação de Laplace, equação de calor (Difusão), equação de Black - Scholes e a equação Fisher-KPP. O método usa diferenças finitas para expressar a equação como um problema de otimização que pode ser resolvido via Simulated Annealing. Os resultados numéricos obtidos pelo método proposto são bastante satisfatórios para uma ampla gama de casos específicos. O método converge para quase todos os casos, além disso, o método é relativamente fácil de implementar e gera aproximações razoavelmente precisas.

Palavras-chave: Simulated Annealing, equações diferenciais parciais, equação de Laplace, equação de calor, métodos numéricos.

Abstract

Melba Luz Torres Ortiz **Simulated Annealing method for solving Partial Differential Equations**. 2017. 19 f. Dissertação (Mestrado) - Instituto de Matemática e Estatística, Universidade de São Paulo, São Paulo, 2017.

We propose a numerical method to solve the partial differential equations, in particular, Laplace Equation, Heat Equation (Diffusion), Black and Scholes and Fisher-KPP Equation. The method uses finite differences to express the equation as an optimization problem that can be solved via Simulated Annealing. The numerical results obtained by the proposed method are quite satisfactory for a wide range of specific cases. The method converges for almost all cases, moreover, the method is relatively easy to implement and generates reasonably accurate approximations.

Keywords: Simulated Annealing, Differential Partial Equations, Laplace equation, Heat equation, numerical method.

Sumário

Lista de Figuras	ix
1 O algoritmo de Simulated Annealing	1
1.1 Simulated Annealing Básico	2
1.2 Funções para otimização via SA	4
1.3 Variantes do Simulated Annealing	5
2 Soluções Numéricas de Equações Diferenciais via Simulated Annealing	7
2.1 Equações Diferenciais Ordinárias (EDO's)	7
2.1.1 Solução Analítica de EDO's de Primeira Ordem	7
2.1.2 Solução Numérica de EDO's de Primeira Ordem	7
2.2 Equações Diferenciais Parciais (EDP's)	8
2.2.1 Solução Numérica de EDP's	10
2.3 Conclusões	14
Referências Bibliográficas	17

Lista de Figuras

1.1	$f(x, y) = -\cos(x)\cos(y)\exp(-(x-\pi)^2 - (y-\pi)^2)$. A função Easom possui vários mínimos locais. É unimodal, e o mínimo global tem uma pequena área relativa ao suporte da função. A função geralmente é avaliada no quadrado $x, y \in [-100, 100]$. A função tem mínimo global de -1 em (π, π) , i.e. $f(\pi, \pi) = -1$	4
1.2	$f(\mathbf{x}) = 10d + \sum_{i=1}^d [x_i^2 - 10\cos(2\pi x_i)]$. A função Rastrigin tem vários mínimos locais. É altamente multimodal, mas os locais dos mínimos são distribuídos regularmente. É mostrado no gráfico acima em sua forma bidimensional. A função é geralmente avaliada no hipercubo $x_i \in [-5.12, 5.12]$ para todo $i = 1, \dots, d$. A função tem mínimo global de 0 em zero, i.e. $f(\mathbf{x}) = 0$ para $\mathbf{x} = (0, \dots, 0)$	5
1.3	$f(\mathbf{x}) = \frac{1}{2} \sum_{i=1}^d (x_i^4 - 16x_i^2 + 5x_i)$. A função Styblinski-Tang é mostrada aqui em sua forma bidimensional. A função é geralmente avaliada no hipercubo $x_i \in [-5, 5]$ para todo $i = 1, \dots, d$. A função tem mínimo global que depende da dimensão, i.e. $f(\mathbf{x}) = -39, 16599d$ para $\mathbf{x} = (2, 903534, \dots, 2, 903534)$	6
2.1	A EDO planteada é $\frac{du}{dt} = 2t + 1$, $u(0) = 2$ para $t \in [0, 1]$. Os pontos vermelhos são a solução aproximada, a linha azul é a solução exata. Os parâmetros usado no SA são: $N = 10$, $\delta_v = 0.005$, $N_{simul} = 3000$, $T_{inicial} = 90$, $\alpha = 0.9$	9
2.2	A EDP planteada é $\frac{\partial^2 u}{\partial t^2} + \frac{\partial^2 u}{\partial x^2} = 0$ com condições de borde $u(t, 0) = 0$. Os parâmetros usado no SA são: $M_t = 8$, $M_x = 10$, $\delta_v = 0.005$, $N_{simul} = 1500$, $T_{inicial} = 20$, $\alpha = 0.99$	11
2.3	Função de Energia H da equação de Laplace.	11
2.4	A EDP planteada é $\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2}$ com condições de borde $u(0, x) = 5x$ e $u(1, x) = 5 + 5x$. Os parâmetros usado no SA são: $M_t = 10$, $M_x = 10$, $\delta_v = 0.005$, $N_{simul} = 1500$, $T_{inicial} = 20$, $\alpha = 0.99$	12
2.5	Função de Energia H da equação de Calor	12
2.6	A EDP é $\frac{\partial u}{\partial t} - 10\frac{\partial^2 u}{\partial x^2} = 0.1u(1-u)$. Os parâmetros usado no SA são: $M_t = 20$, $M_x = 20$, $\delta_v = 0.001$, $N_{simul} = 10000$, $T_{inicial} = 40$, $\alpha = 0.99$	13
2.7	A EDP é $\frac{\partial u}{\partial t} - 10\frac{\partial^2 u}{\partial x^2} = 0.2u(1-u)$. Os parâmetros usado no SA são: $M_t = 20$, $M_x = 20$, $\delta_v = 0.001$, $N_{simul} = 10000$, $T_{inicial} = 40$, $\alpha = 0.99$	14

2.8 A EDP é $\frac{\partial u}{\partial t} - \frac{\partial^2 u}{\partial x^2} = 0.3u(1 - u)$. Os parâmetros usado no SA são: $M_t = 20$,
 $M_x = 20$, $\delta_v = 0.001$, $N_{simul} = 10000$, $T_{inicial} = 40$, $\alpha = 0.99$ 15

Capítulo 1

O algoritmo de Simulated Annealing

O annealing (recozimento) é referido ao temperamento de certas ligas de metal, vidro ou cristal por aquecimento acima do seu ponto de fusão, mantendo a temperatura e depois esfriando-o muito lentamente até se solidificar em uma perfeita estrutura cristalina. Este processo físico-químico produz materiais de alta qualidade. A simulação deste processo é conhecida como Simulated Annealing (SA) [V85, SJM83]. O estado de cristal livre de defeito corresponde à configuração de energia mínima global. Existe uma analogia de SA com um procedimento de otimização. Os estados materiais físicos correspondem a soluções do problema, a energia de um estado a custo de uma solução e a temperatura a um parâmetro de controle.

O algoritmo Metropolis é um método simples para simular a evolução do equilíbrio térmico de um sólido para uma determinada temperatura [NAM⁺53]. SA [SJM83] é uma variante do algoritmo Metropolis, onde a temperatura está mudando de alta para baixa. SA é composta basicamente de dois processos estocásticos: um processo para a geração de soluções e outro para a aceitação de soluções. A temperatura de geração é responsável pela correlação entre as possíveis soluções geradas e a solução original.

SA é um algoritmo de descida modificado por movimentos de ascensão aleatórios para escapar dos mínimos locais e vá na procura de mínimos globais. O algoritmo annealing simula uma cadeia de Markov de estado finito não estacionário cujo espaço de estado é o domínio da função a ser minimizado. A Amostragem de importância é o principal princípio que está subjacente à SA. Ele foi usado na física estatística para escolher estados de amostra de um modelo de sistema de partículas para estimar de forma eficiente algumas quantidades físicas. A amostragem por Importância favorece estados com energias mais baixas.

SA é um algoritmo serial de propósito geral para encontrar um mínimo global para uma função contínua. Também é um algoritmo de Monte Carlo convencional para qualquer problema de otimização. As soluções por esta técnica estão próximas do mínimo global dentro de um limite superior polinomial para o tempo computacional e são independentes das condições iniciais. Alguns algoritmos paralelos para SA foram propostos visando melhorar a precisão das soluções aplicando paralelismo [ZJ.01].

1.1 Simulated Annealing Básico

De acordo com a termodinâmica estatística, P_α , a probabilidade de um sistema físico estar no estado α com energia E_α à temperatura T satisfaz a distribuição de Boltzmann.

$$P_\alpha = \frac{1}{Z} e^{-\frac{E_\alpha}{K_B T}}, \quad (1.1)$$

onde K_B é a constante de Boltzmann's, T é temperatura absoluta, e Z é a função participação, definido por

$$Z = \sum_{\beta} e^{-\frac{E_\beta}{K_B T}}, \quad (1.2)$$

a soma de todos os estados β com energia E_β à temperatura T . Para T grande, a distribuição de Boltzmann exibe preferência uniforme para todos os estados, independentemente da energia. Quando T se aproxima para zero, apenas os estados com energia mínima têm probabilidade de ocorrência diferente de zero.

Em SA, a constante K_B é omitido. Para T grande, o sistema ignora pequenas mudanças na energia e se aproxima ao equilíbrio térmico rapidamente, ou seja, realiza uma pesquisa do espaço dos estados globais e encontra um mínimo. À medida que T é abaixado, o sistema responde a pequenas mudanças na energia e realiza uma busca fina na vizinhança do mínimo já determinado e encontra um mínimo melhor. Em $T = 0$, qualquer alteração nos estados do sistema não leva a um aumento na energia e, portanto, o sistema deve alcançar o equilíbrio se $T = 0$.

Ao realizar SA, teoricamente, um mínimo global é garantido para ser alcançado com alta probabilidade. O ruído térmico artificial diminui gradualmente no tempo. T é um parâmetro de controle chamado temperatura computacional, que controla a magnitude das perturbações da função de energia $E(x)$. A probabilidade de uma mudança de estado é determinada pela distribuição de Boltzmann da diferença de energia dos dois estados:

$$P = e^{-\frac{\Delta E}{T}}. \quad (1.3)$$

A probabilidade de se mover para cima na função de energia ($\Delta E > 0$) é grande para T grande, e é baixa para T baixo. SA permite movimentos ascendentes de forma controlada: tenta melhorar a procura local, ocasionalmente, assumindo um risco e aceitando uma solução pior. SA pode ser realizado como Algoritmo 1 [SJM83].

Algorithm 1 Algoritmo SA

 1. Inicialize a configuração do sistema.

 $\mathbf{x}(0)$ aleatório.

 2. Inicialize T e T_{min} com um valor grande e pequeno respectivamente.

 3. Fixe o número de simulações por temperatura N_{simul} .

 4. ENQUANTO $T < T_{min}$ Inicialize $i = 0$

 ENQUANTO $i < N_{simul}$

 i. Aplicar perturbações aleatórias ao estado $\mathbf{x} = \mathbf{x} + \Delta\mathbf{x}$.

 ii. Avalie $\Delta E(\mathbf{x}) = E(\mathbf{x} + \Delta\mathbf{x}) - E(\mathbf{x})$:

 SENÃO $\Delta E(\mathbf{x}) < 0$ mantenha o novo estado aceite o novo estado com probabilidade
 $P = e^{-\frac{\Delta E}{T}}$. $i = i + 1$ $T = T - \Delta T$

O procedimento básico de SA é conhecido como Boltzmann annealing. A sequência do resfriamento é fundamental para a eficiência da SA. Se T for reduzido muito rapidamente, a convergência prematura para um mínimo local pode ocorrer. Em contraste, se é muito lento, o algoritmo é muito lento para convergir. Com base em uma análise da cadeia Markov no processo SA, Geman e Geman [SD84] provaram que uma condição simples e necessária suficiente na sequência de resfriamento para que o algoritmo converge em probabilidade para o conjunto de estados de mínimo global é que T deve ser diminuído de acordo com

$$T(t) \geq \frac{T_0}{\ln(1+t)}, \quad t = 1, 2, \dots \quad (1.4)$$

para garantir a convergência para o mínimo global com a probabilidade um, onde T_0 é uma temperatura inicial suficientemente grande. Dado um número suficientemente grande de iterações a cada temperatura, é provado que a SA converge quase que certamente para o ótimo global [B.88]. Em [B.88], T_0 é provado ser maior ou igual à profundidade do mínimo local mais profundo que não é o estado mínimo global. A fim de garantir Boltzmann annealing convergir para o mínimo global com a probabilidade um, $T(t)$ precisa diminuir logaritmicamente com o tempo.

1.2 Funções para otimização via SA

A maior vontade do SA é que ela não fica atrapalhado nos ótimos locais e vai na procura de ótimos globais, isso faz que ela funcione muito bem na otimização de funções não convexas, além disso a natureza de pulos no conjunto de restrições faz que ela seja aplicável a problemas de otimização combinatória. Apresentaremos algumas funções candidatas onde SA teria um ótimo performance.

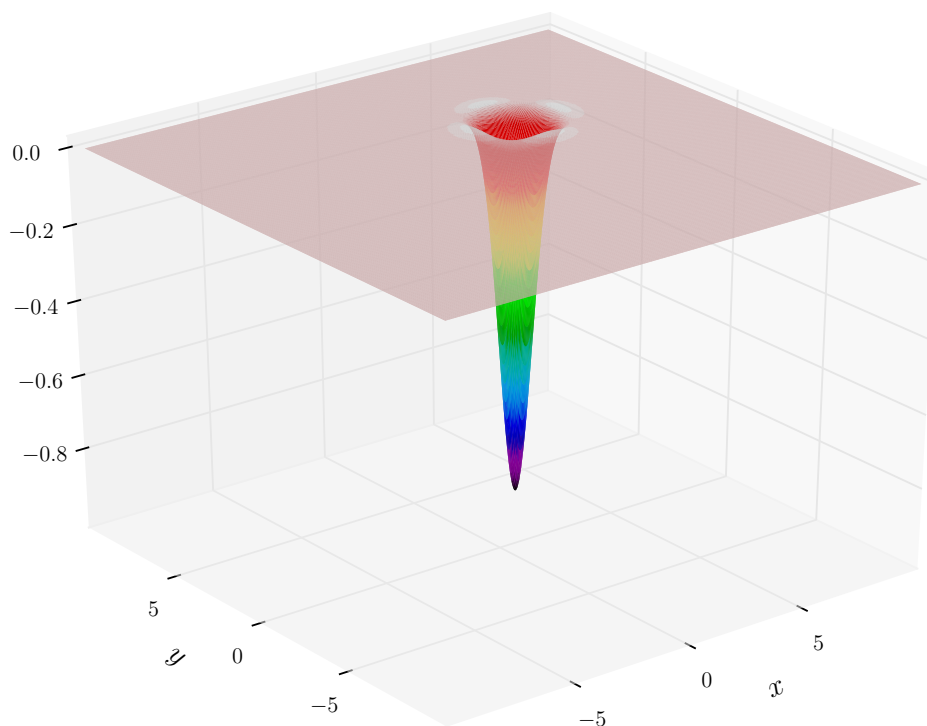


Figura 1.1: $f(x, y) = -\cos(x)\cos(y)\exp(-(x-\pi)^2 - (y-\pi)^2)$. A função *Easom* possui vários mínimos locais. É unimodal, e o mínimo global tem uma pequena área relativa ao suporte da função. A função geralmente é avaliada no quadrado $x, y \in [-100, 100]$. A função tem mínimo global de -1 em (π, π) , i.e. $f(\pi, \pi) = -1$.

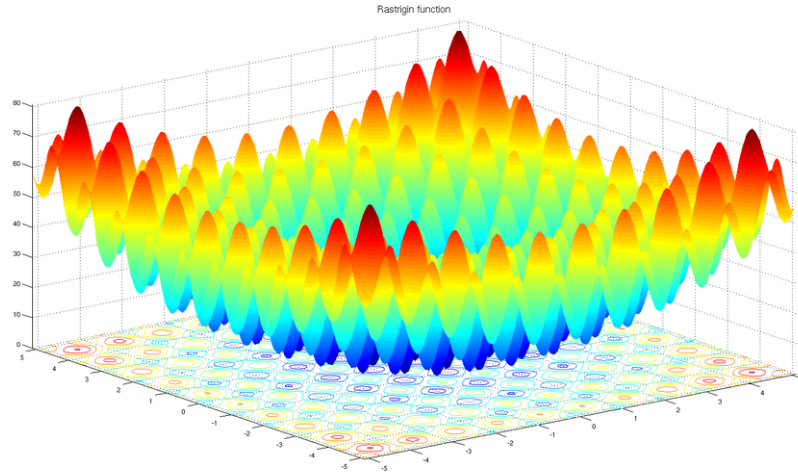


Figura 1.2: $f(\mathbf{x}) = 10d + \sum_{i=1}^d [x_i^2 - 10 \cos(2\pi x_i)]$. A função Rastrigin tem vários mínimos locais. É altamente multimodal, mas os locais dos mínimos são distribuídos regularmente. É mostrado no gráfico acima em sua forma bidimensional. A função é geralmente avaliada no hipercubo $x_i \in [-5.12, 5.12]$ para todo $i = 1, \dots, d$. A função tem mínimo global de 0 em zero, i.e. $f(\mathbf{x}) = 0$ para $\mathbf{x} = (0, \dots, 0)$.

1.3 Variantes do Simulated Annealing

Standard SA é um método de busca estocástica e a convergência para o ótimo global é muito lenta para uma sequência de resfriamento confiável. Muitos métodos, como o Cauchy annealing [HR87], simulated reannealing [L.89], SA generalizada [CD96] e SA com valor global conhecido [M.01] foram propostos para acelerar a procura de SA. Existem também métodos globais de otimização que utilizam a idéia de annealing [JFC98, K.98].

O Cauchy annealing [HR87] substitui a distribuição de Boltzmann pela distribuição de Cauchy, também conhecida como a distribuição de Cauchy-Lorentz. A variância infinita proporciona uma melhor capacidade de escapar dos mínimos locais e permite o uso de sequências mais rápidas, como T diminuindo de acordo com $T(t) = \frac{T_0}{t}$.

Na simulated reannealing [L.89], T diminui exponencialmente com t :

$$T = T_0 e^{-\frac{c_1 t}{J}}, \quad (1.5)$$

onde a constante $c_1 > 0$ e J é a dimensão do espaço de entrada. A introdução da reannealing também permite a adaptação às insensibilidades em mudança no espaço de parâmetros multidimensionais.

A SA generalizada [CD96], generaliza o Cauchy annealing [HR87] e o Boltzmann annealing [SJM83] dentro de um quadro unificado inspirado nas termostáticas generalizadas. Oposição-baseada SA [MH07], melhora SA em precisão e taxa de convergência usando vizinhos opostos.

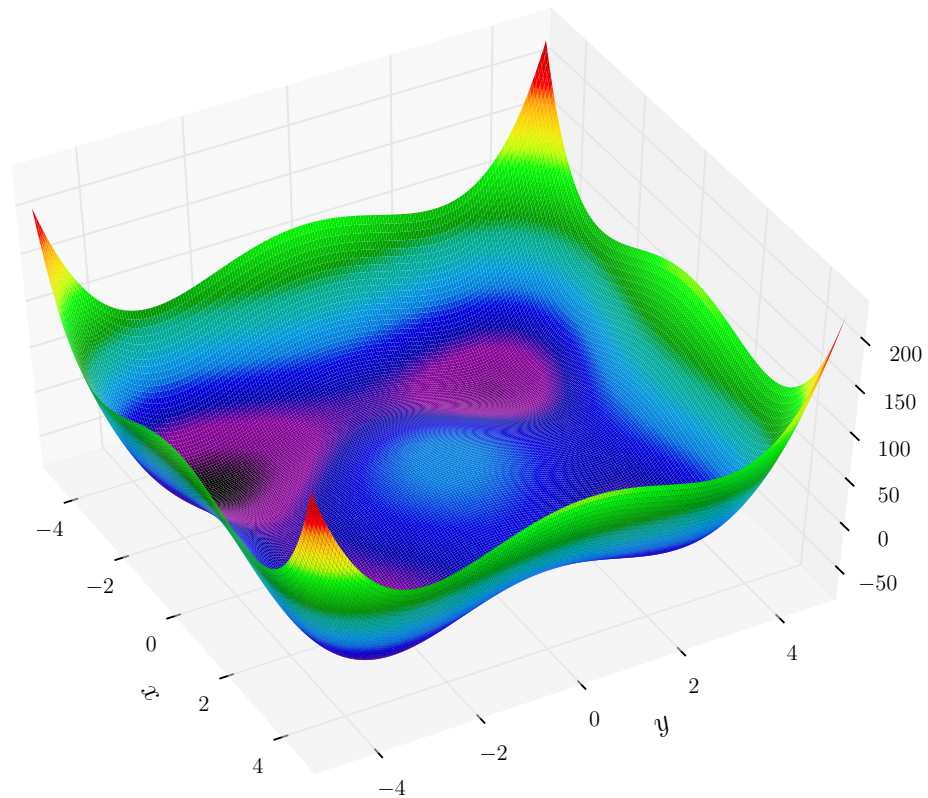


Figura 1.3: $f(\mathbf{x}) = \frac{1}{2} \sum_{i=1}^d (x_i^4 - 16x_i^2 + 5x_i)$. A função *Styblinski-Tang* é mostrada aqui em sua forma bidimensional. A função é geralmente avaliada no hipercubo $x_i \in [-5, 5]$ para todo $i = 1, \dots, d$. A função tem mínimo global que depende da dimensão, i.e. $f(\mathbf{x}) = -39,16599d$ para $\mathbf{x} = (2, 903534, \dots, 2, 903534)$.

Capítulo 2

Soluções Numéricas de Equações Diferenciais via Simulated Annealing

Problemas na física, biologia, Finanças que envolvem mais de uma variável são muitas vezes expresso utilizando equações diferenciais, entre mais complexo o fenômeno fica mais difícil obter soluções analíticas recorrendo na maioria dos casos a soluções numéricas. Nesta seção, apresentamos como o Simulated Annealing pode ser usada para obter soluções numéricas de equações diferenciais em contraste com métodos convencionais como o método de diferenças finitas que exigem condições mais restritas.

2.1 Equações Diferenciais Ordinárias (EDO's)

2.1.1 Solução Analítica de EDO's de Primeira Ordem

EDO de primeira ordem geral

$$\frac{du}{dt} = f(t, u), \quad u(t_0) = u_0 \quad (2.1)$$

O método de separação de variáveis pode ser aplicado para resolver casos simples. Quando as duas variáveis contribuem separadamente, ou seja, $f(t, u) = G(t)H(u)$, o ODE acima pode ser escrito como:

$$\frac{du}{H(u)} = \frac{dt}{G(t)}, \quad u(t_0) = u_0 \quad (2.2)$$

Após a integração em ambos os lados, a solução é:

$$\int_{u_0}^u \frac{du}{H(u)} = \int_{t_0}^t \frac{dt}{G(t)} \quad (2.3)$$

2.1.2 Solução Numérica de EDO's de Primeira Ordem

Considere um problema de valor inicial do tipo

$$\frac{du}{dt} = f(t), \quad u(t_0) = u_0 \quad (2.4)$$

onde f é uma função geral da própria solução, e u_0 é um valor conhecido.

A ideia geral para solucionar uma equação diferencial via SA é procurar uma função de desvio entre a discretização de $f(t, u)$ e $\frac{du}{dt}$, tal que a minimizar a função de desvio que atinga u_0 .

Para lo qual, na equação substituímos a formula de derivação numérica diferença finita centrada, temos que:

$$\frac{u(t+h) - u(t-h)}{2h} = f(t), \quad h \rightarrow 0 \quad (2.5)$$

Seja o intervalo de solução $[a, b]$. Em concordância com as condições iniciais e o intervalo da solução se escolhe h e porem a fica escolhido a discretização do domínio. Considere N como o tamanho da discreteação do intervalo da solução.

Seja $t_{i+1} = t_i + h$ para $i = 1, 2, \dots, N-1$. A função objetivo H que sera minimizada via SA sera definida da seguinte forma:

$$H = \sum_{i=1}^{N-1} \left[\frac{u(t_{i+1}) - u(t_{i-1}))}{2h} - f(t_i) \right]^2 \quad (2.6)$$

A minimização da função H via SA, sera o critério para a escolha da melhor solução aproximada. Deve-se notar que H representa a energia ou hamiltoniano do sistema SA vai procurar o estado de menor energia. A figura 2.1 apresenta um exemplo numero para o caso onde f é linear.

Podemos observar que a solução aproximada é bastante consistente com a solução teórica. Um procedimento muito parecido pode ser usado para casos mais gerais ou de maior ordem.

2.2 Equações Diferenciais Parciais (EDP's)

Quase todas as EDPs usadas para descrever numericamente os fenômenos na Física, Biologia e Finanças são lineares e de segunda ordem. Uma EDP linear de segunda ordem satisfeita por uma função $u(x, t)$ dependendo de apenas duas variáveis (espaço e tempo):

$$A \frac{\partial^2 u}{\partial t^2} + 2B \frac{\partial^2 u}{\partial x \partial t} + C \frac{\partial^2 u}{\partial x^2} = D(t, x, u, \frac{\partial u}{t}, \frac{\partial u}{x}) \quad (2.7)$$

onde D é uma função linear em $\frac{\partial u}{t}$ e $\frac{\partial u}{x}$. Podemos classificar as EDP's em tres categorias:

1. Hiperbólico ($B^2 - AC > 0$)

Exemplo: Equação de onda - dimensão 1

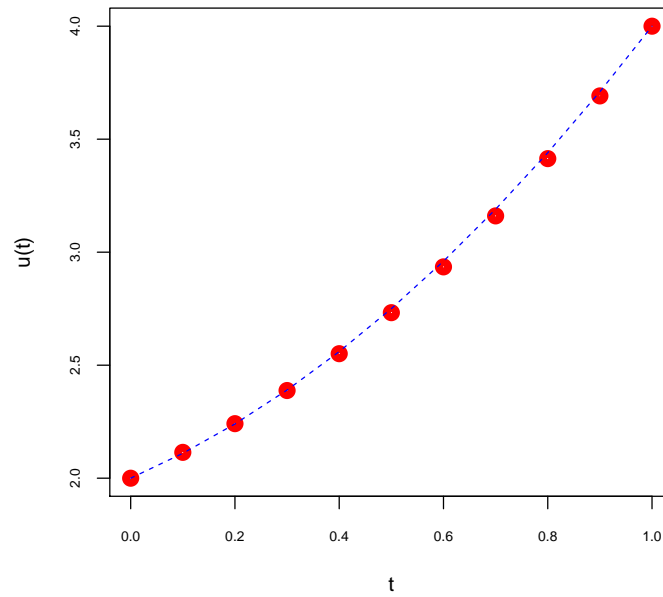


Figura 2.1: A EDO planteada é $\frac{du}{dt} = 2t + 1$, $u(0) = 2$ para $t \in [0, 1]$. Os pontos vermelhos são a solução aproximada, a linha azul é a solução exata. Os parâmetros usado no SA são: $N = 10$, $\delta_v = 0.005$, $N_{simul} = 3000$, $T_{inicial} = 90$, $\alpha = 0.9$

$$\frac{\partial^2 u}{\partial x^2} = \frac{1}{c^2} \frac{\partial^2 u}{\partial t^2} \quad (2.8)$$

2. Parabólico ($B^2 - AC = 0$)

Exemplo: Equação de Difusão (Black Sholes)

$$\frac{\partial u}{\partial t} = D \frac{\partial^2 u}{\partial t^2} \quad (2.9)$$

3. Elíptico ($B^2 - AC < 0$)

Exemplo: Equação de Laplace

$$\frac{\partial^2 u}{\partial t^2} = \frac{\partial^2 u}{\partial x^2} \quad (2.10)$$

2.2.1 Solução Numérica de EDP's

Equação de Laplace

Considere a seguinte expressão para la equação de Laplace

$$\frac{\partial^2 u}{\partial t^2} + \frac{\partial^2 u}{\partial x^2} = 0 \quad (2.11)$$

Seguindo o mesmo procedimento que no caso de uma EDO. Substituímos a formula da segunda derivação numérica, temos que:

$$\frac{u(t + h_t) - 2u(t) + u(t - h_t)}{h_t^2} + \frac{u(x + h_x) - 2u(x) + u(x - h_x)}{h_x^2} = 0 \quad (2.12)$$

Seja a caixa $[a, b] \times [c, d]$ o conjunto solução. Em concordância com as condições iniciais e o conjunto de solução se escolhe valores pequenos para h_t e h_x . Porem a fica escolhido a discretização do domínio. Considere N_t e N_x como o tamanho da discreteação do intervalo $[a, b]$ e $[c, d]$ respectivamente.

Seja $t_{i+1} = t_i + h_t$ para $i = 1, \dots, N_t - 1$, e seja $x_{j+1} = x_j + h_x$ para $j = 1, \dots, N_x - 1$. A função objetivo H que sera minimizada via SA sera definida da seguinte forma:

$$H = \sum_{j=1}^{N_x-1} \sum_{i=1}^{N_t-1} \left[\frac{u(t_{i+1}) - 2u(t_i) + u(t_{i-1}))}{h_t^2} + \frac{u(x_{j+1}) - 2u(x_j) + u(x_{j-1}))}{h_x^2} \right]^2 \quad (2.13)$$

Na figura 2.2 se apresenta uma solução numérica para la equação de Laplace e na figura 2.3 se apresenta o decaimento da função H que da informação da convergência do método SA na solução da equação.

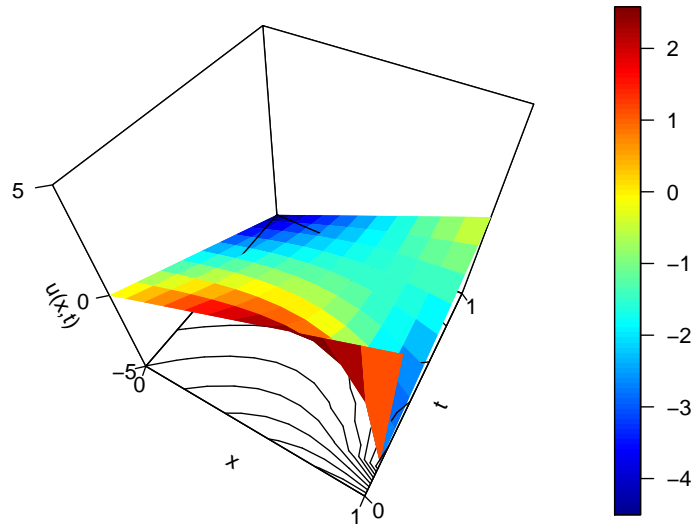


Figura 2.2: A EDP planteada é $\frac{\partial^2 u}{\partial t^2} + \frac{\partial^2 u}{\partial x^2} = 0$ com condições de borde $u(t, 0) = 0$. Os parâmetros usado no SA são: $M_t = 8$, $M_x = 10$, $\delta_v = 0.005$, $N_{simul} = 1500$, $T_{inicial} = 20$, $\alpha = 0.99$.

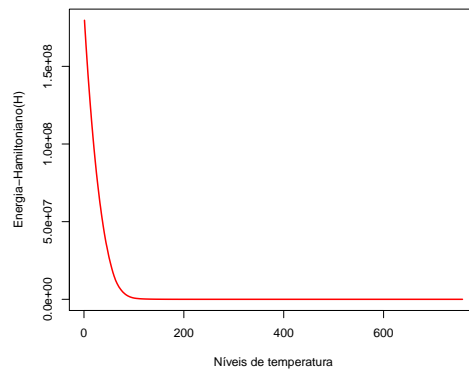


Figura 2.3: Função de Energia H da equação de Laplace.

Equação de Calor

Considere a seguinte expressão para a equação de calor.

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} \quad (2.14)$$

A função objetivo H neste caso sera definido como segue

$$H = \sum_{j=1}^{N_x-1} \sum_{i=1}^{N_t-1} \left[\frac{u(t_{i+1}) - u(t_{i-1})}{2h_t} - \frac{u(x_{j+1}) - 2u(x_j) + u(x_{j-1}))}{h_x^2} \right]^2 \quad (2.15)$$

Na figura 2.4 e 2.5 se apresenta resultados da solução numérica da equação de calor.

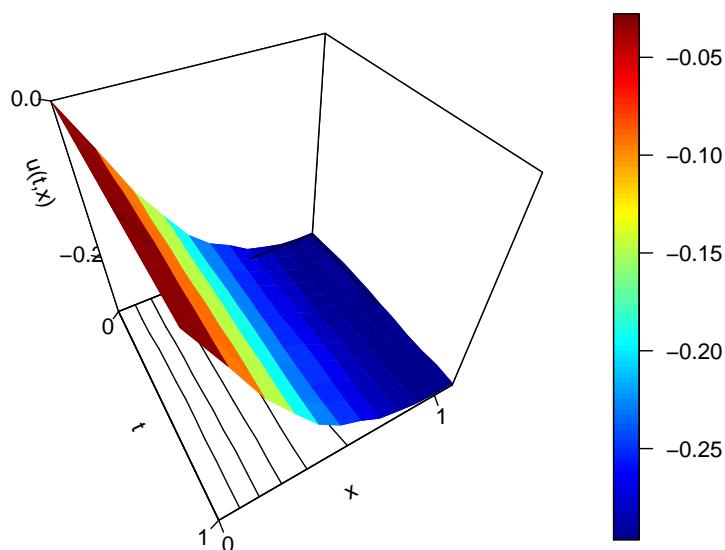


Figura 2.4: A EDP planteada é $\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2}$ com condições de borde $u(0, x) = 5x$ e $u(1, x) = 5 + 5x$. Os parâmetros usado no SA são: $M_t = 10$, $M_x = 10$, $\delta_v = 0.005$, $N_{simul} = 1500$, $T_{inicial} = 20$, $\alpha = 0.99$.

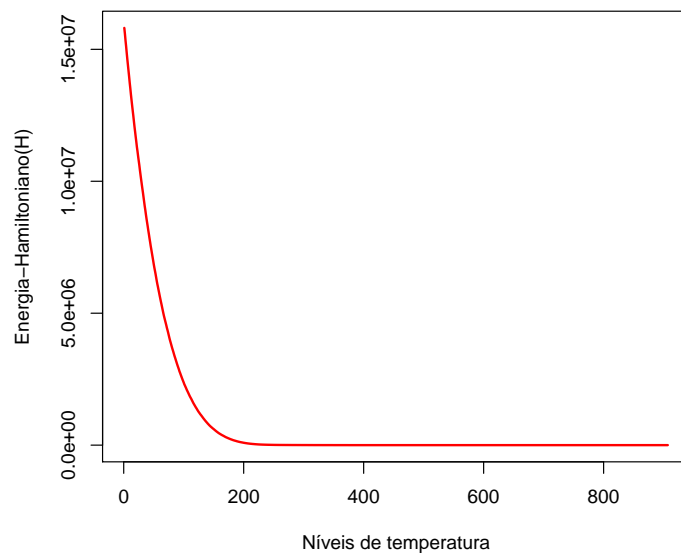


Figura 2.5: Função de Energia H da equação de Calor

Equação Fisher-KPP

Em matemática, a equação de Fisher (que recebeu o nome do estatístico e do biólogo Ronald Fisher, também conhecida como equação de Kolmogorov-Petrovsky-Piskunov - cha-

mada asem por Andrey Kolmogorov, Ivan Petrovsky e N. Piskunov - ou equação Fisher-KPP) é a equação diferencial parcial :

$$\frac{\partial u}{\partial t} - D \frac{\partial^2 u}{\partial x^2} = ru(1 - u) \quad (2.16)$$

A função objetivo H neste caso sera definido como segue

$$H = \sum_{j=1}^{N_x-1} \sum_{i=1}^{N_t-1} \left[\frac{u(t_{i+1}) - u(t_{i-1}))}{2h_t} - D \frac{u(x_{j+1}) - 2u(x_j) + u(x_{j-1}))}{h_x^2} - ru(t_i, x_j)(1 - u(t_i, x_j)) \right]^2 \quad (2.17)$$

Nas seguintes figuras apresentamos resultados numéricos para alguns valores de los parâmetros da equação.

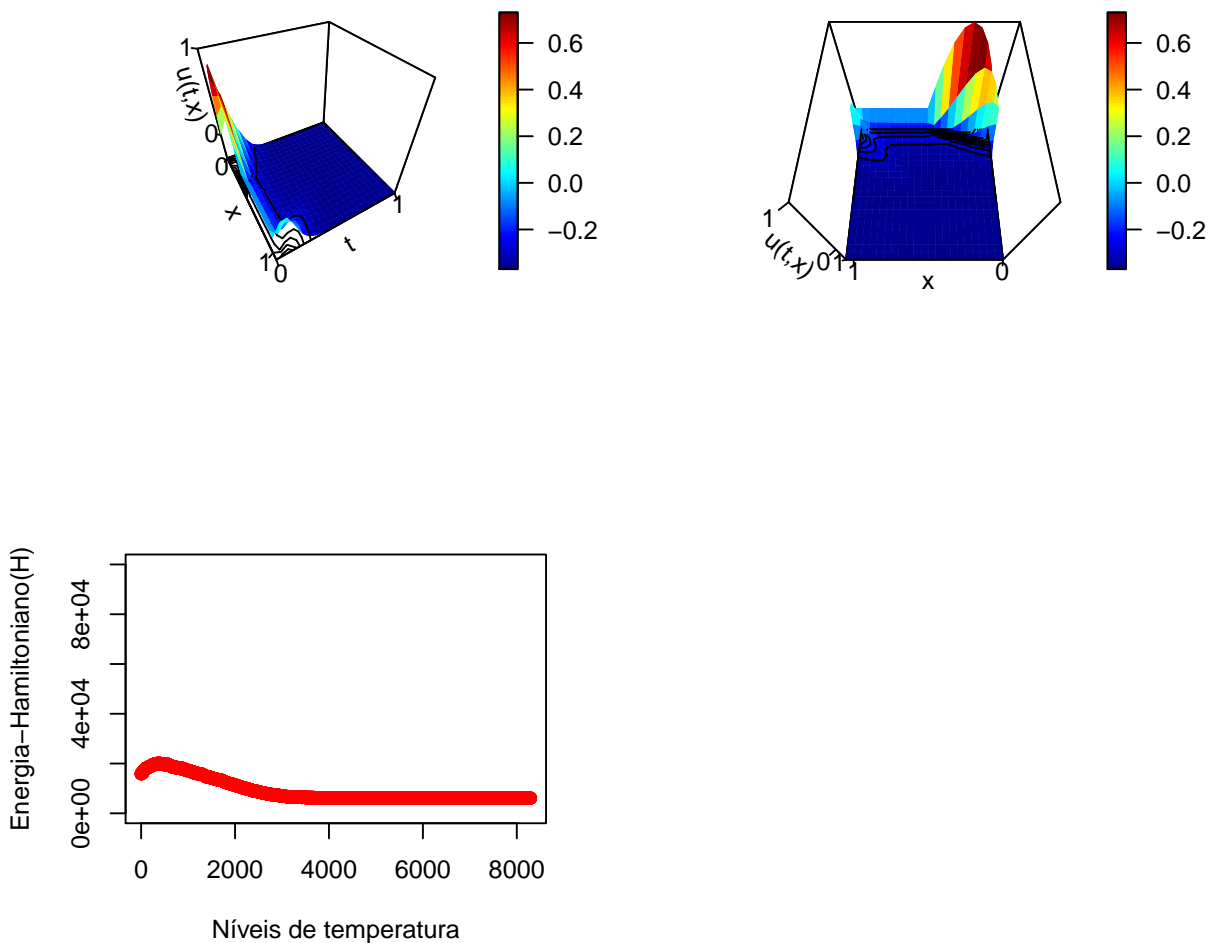


Figura 2.6: A EDP é $\frac{\partial u}{\partial t} - 10 \frac{\partial^2 u}{\partial x^2} = 0.1u(1 - u)$. Os parâmetros usado no SA são: $M_t = 20$, $M_x = 20$, $\delta_v = 0.001$, $N_{simul} = 10000$, $T_{inicial} = 40$, $\alpha = 0.99$.

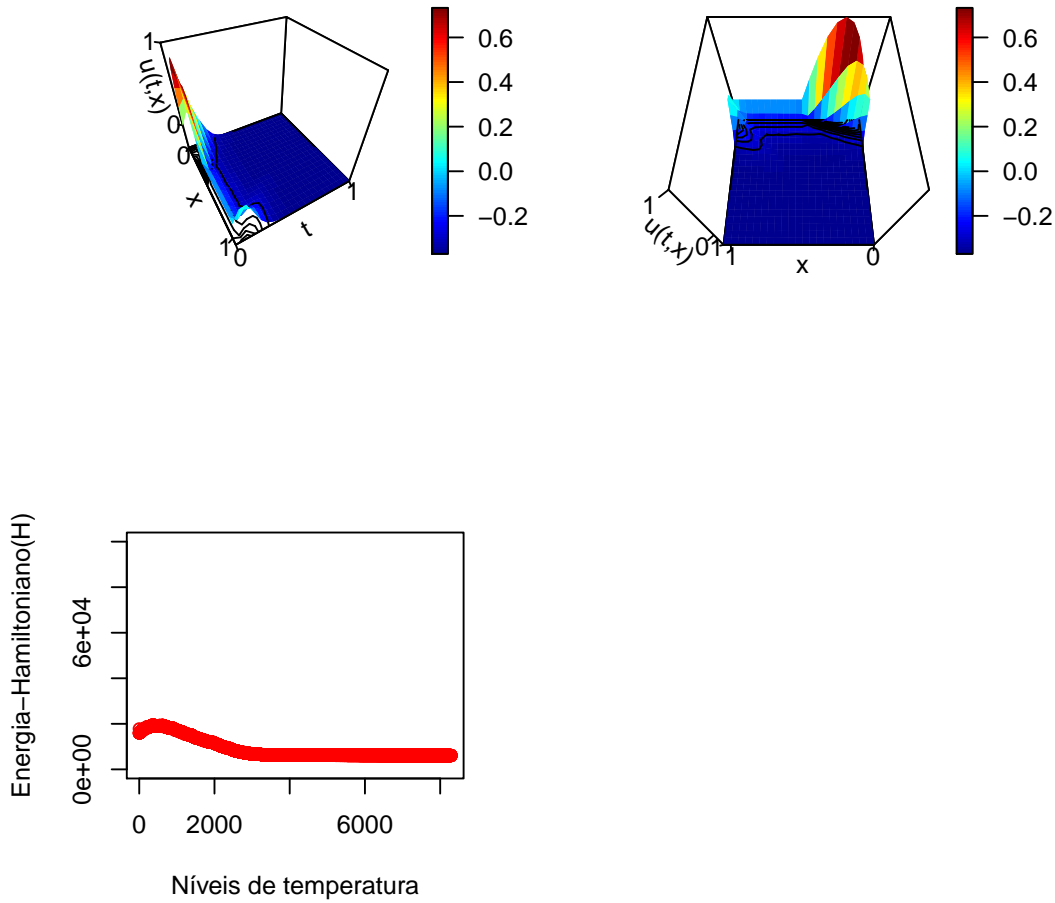


Figura 2.7: A EDP é $\frac{\partial u}{\partial t} - 10 \frac{\partial^2 u}{\partial x^2} = 0.2u(1-u)$. Os parâmetros usado no SA são: $M_t = 20$, $M_x = 20$, $\delta_v = 0.001$, $N_{simul} = 10000$, $T_{inicial} = 40$, $\alpha = 0.99$.

2.3 Conclusões

O método numérico proposto para resolver as equações EDO linear, Equação de Laplace, Equação de Calor(Difusão) e a equação de Fisher-KPP usando diferenças finitas para expressar a equação como problema de otimização que pode ser resolvido via Simulated Annealing. Os resultados numéricos obtidos pelo método proposto são bastante satisfatórios para uma ampla gama de casos específicos. O método converge para quase a totalidade dos casos, além disso, o método é relativamente fácil de implementar e gera aproximações razoavelmente precisas. A estabilidade e a consistência do método proposto pode ser tópico para futuras pesquisas.

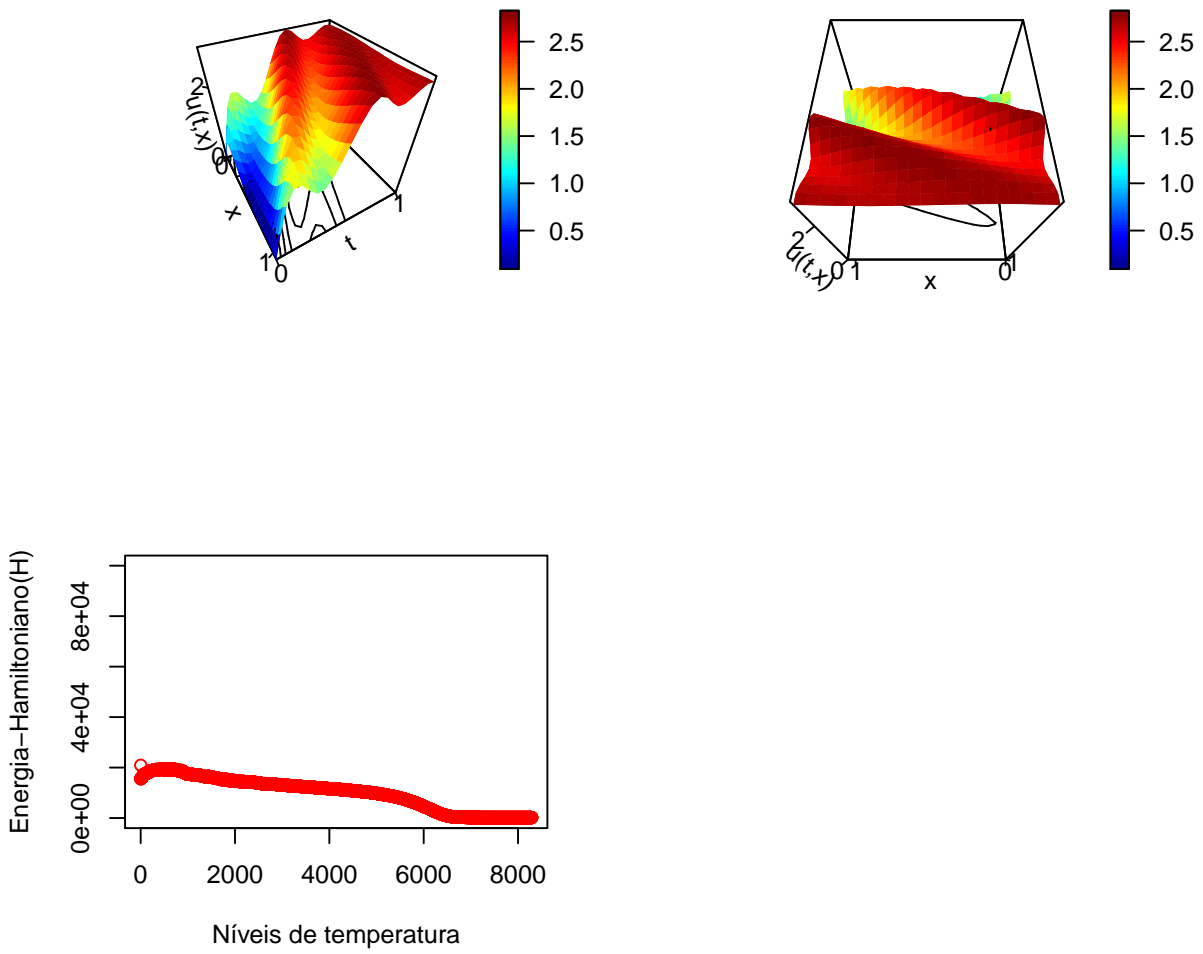


Figura 2.8: A EDP é $\frac{\partial u}{\partial t} - \frac{\partial^2 u}{\partial x^2} = 0.3u(1-u)$. Os parâmetros usado no SA são: $M_t = 20$, $M_x = 20$, $\delta_v = 0.001$, $N_{simul} = 10000$, $T_{inicial} = 40$, $\alpha = 0.99$.

Referências Bibliográficas

- [B.88] Hajek B. Cooling schedules for optimal annealing. *Math Oper Res.*, pages 311–29, 1988. 3
- [CD96] Tsallis C and Stariolo DA. Generalized simulated annealing. *Phys A.*, pages 395–406, 1996. 5
- [HR87] Szu HH and Hartley RL. Nonconvex optimization by fast simulated annealing. *Proc IEEE.*, pages 1538–40, 1987. 5
- [JFC98] Richardt J, Karl F and Muller C. Connections between fuzzy theory, simulated annealing, and convex duality. *Fuzzy Sets Syst.*, pages 307–34, 1998. 5
- [K.98] Rose K. Deterministic annealing for clustering, compression, classification, regression, and related optimization problems. *Proc IEEE.*, pages 2210–39, 1998. 5
- [L.89] Ingber L. Very fast simulated re-annealing. *Math Comput Model.*, pages 967–73, 1989. 5
- [M.01] Locatelli M. Convergence and rst hitting time of simulated annealing algorithms for continuous global optimization. *Math Methods Oper Res.*, pages 171–99, 2001. 5
- [MH07] Ventresca M and Tizhoosh HR. Simulated annealing with opposite neighbors. *Proceedings of the IEEE symposium on foundations of computational intelligence (SIS 2007)*, pages 186–192, 2007. 5
- [NAM⁺53] Metropolis N, Rosenbluth A, Rosenbluth M, Teller A and Teller E. Equations of state calculations by fast computing machines. *J Chem Phys.*, pages 1087–92, 1953. 1
- [SD84] Geman S and Geman D. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE Trans Pattern Anal Mach Intell*, pages 721–41, 1984. 3
- [SJM83] Kirkpatrick S, Gelatt CD Jr and Vecchi MP. Optimization by simulated annealing. *Science*, pages 671–80, 1983. 1, 2, 5
- [V85] Cerny V. Thermodynamical approach to the traveling salesman problem: an efficient simulation algorithm. *J Optim Theory Appl.*, pages 45: 41–51, 1985. 1
- [ZJ.01] Czech ZJ. *Three parallel algorithms for simulated annealing*. Proceedings of the 4th international conference on parallel processing and applied mathematics. Naczow, Poland.London: Springer, 2001. 1