

Têmpera simulada (*Simulated Annealing*)

Airton Bordin Junior

[airtonbjunior@gmail.com]

Metaheurísticas - Prof. Dr. Celso Gonçalves Camilo Junior

Mestrado em Ciência da Computação 2017/2

Universidade Federal de Goiás (UFG) - Instituto de Informática – Setembro/2017

Programação

- Introdução
- Problema bem definido
- Busca informada e não informada
- *Simulated Annealing*
- Referências





Problema bem definido

- Componentes
 1. Estado inicial;
 2. Ações;
 3. Modelo de transição;
 4. Teste de objetivo
 5. Custo de caminho.
- Ambiente do problema: representado pelo espaço de estados
 - Solução: um caminho do estado inicial ao estado objetivo.



Algoritmos de busca

- Analisados em termos de
 - Completeza;
 - Otimização;
 - Complexidade de tempo;
 - Complexidade de espaço.
- Complexidade
 - b : fator de ramificação no espaço de estados;
 - d : profundidade da solução mais rasa.



Algoritmos de busca

- Analisados em termos de
 - Completeza
 - O algoritmo oferece a garantia de encontrar uma solução quando ela existir?
 - Otimização
 - A estratégia encontra a solução ótima?
 - Complexidade de tempo
 - Tempo necessário para encontrar a solução.
 - Complexidade de espaço.
 - Memória necessária para encontrar a solução.



Busca não informada

- Possuem acesso apenas à definição do problema;
- Algoritmos básicos
 - Busca em largura;
 - Busca de custo uniforme;
 - Busca em profundidade;
 - Busca de aprofundamento iterativo;
 - Busca bidimensional.



Problema bem definido

Quebra-cabeça de oito peças

7	2	4
5		6
8	3	1

Estado inicial

	1	2
3	4	5
6	7	8

Estado objetivo

Estado inicial: Qualquer estado;

Ações: A formulação mais simples define as ações como movimentos do quadrado vazio Esquerda, Direita, Para Cima ou Para Baixo. Pode haver subconjuntos diferentes desses, dependendo de onde estiver o quadrado vazio;

Modelo de transição: Dado um estado e ação, ele devolve o estado resultante; por exemplo, se aplicarmos Esquerda para o estado inicial na figura, o estado resultante terá comutado o 5 e o vazio;

Teste de objetivo: Verifica se o estado corresponde à configuração de estado objetivo mostrada na Figura (são possíveis outras configurações de objetivos);

Custo de caminho: Cada passo custa 1 e, assim, o custo do caminho é o número de passos do caminho.



Busca informada

- Geração de heurísticas admissíveis
 - **Problema relaxado:** problema com poucas restrições sobre as ações;
 - Qualquer solução ótima do problema original será uma solução do problema relaxado;
 - Problema relaxado pode ter melhores soluções
 - Custo de uma solução ótima para um problema relaxado é uma **heurística admissível** para o problema original.



Busca informada

- Bancos de dados de padrões
 - Armazenar os custos exatos de solução para todas as instâncias possíveis do subproblema;
 - Heurística admissível hBD para cada estado completo encontrado durante uma busca
 - Exame da configuração correspondente do subproblema no banco de dados.
 - O próprio banco de dados é construído através de busca reversa do objetivo e do registro do custo de cada novo padrão encontrado..



Busca local e otimização

- Quando o caminho até o objetivo não importa, podemos considerar uma classe diferente de algoritmos
 - Algoritmos de busca local operam usando um único estado atual e, em geral, se movem apenas para os vizinhos desse estado.
- Úteis para **problemas de otimização**
 - Encontrar o melhor estado de acordo com uma função objetivo.



Busca local e otimização

- Vantagens
 - Usam pouquíssima memória (normalmente um valor constante);
 - Frequentemente podem encontrar soluções razoáveis em grandes ou infinitos (contínuos) espaços de estados para os quais os algoritmos sistemáticos são inadequados.



Têmpera simulada

- Kirkpatrick (1983);
- Meta-heurística para otimização: técnica de busca local probabilística - analogia com a termodinâmica;
- Simulação do processo de cozimento de metais;
- Resfriamento rápido produz metais menos estáveis (de maior energia interna) que o resfriamento lento
 - Material passa por diversos estados durante o processo
 - Se houver tempo suficiente, um elemento qualquer do conjunto passa por todos os estados acessíveis



Têmpera simulada

- Analogia com o processo físico
 - Possíveis estados de um metal – soluções no espaço de busca;
 - Energia em cada estado: valor função objetivo;
 - A energia mínima (se o problema for de minimização) - valor de uma solução ótima local, possivelmente global.



Têmpera simulada

- Escolhe um movimento aleatório;
 - Se o movimento melhorar a situação sempre será aceito;
- Caso contrário, o algoritmo aceitará o movimento com alguma probabilidade menor que 1;
 - Probabilidade decresce exponencialmente com a “má qualidade” do movimento — o valor ΔE segundo o qual a avaliação piora.



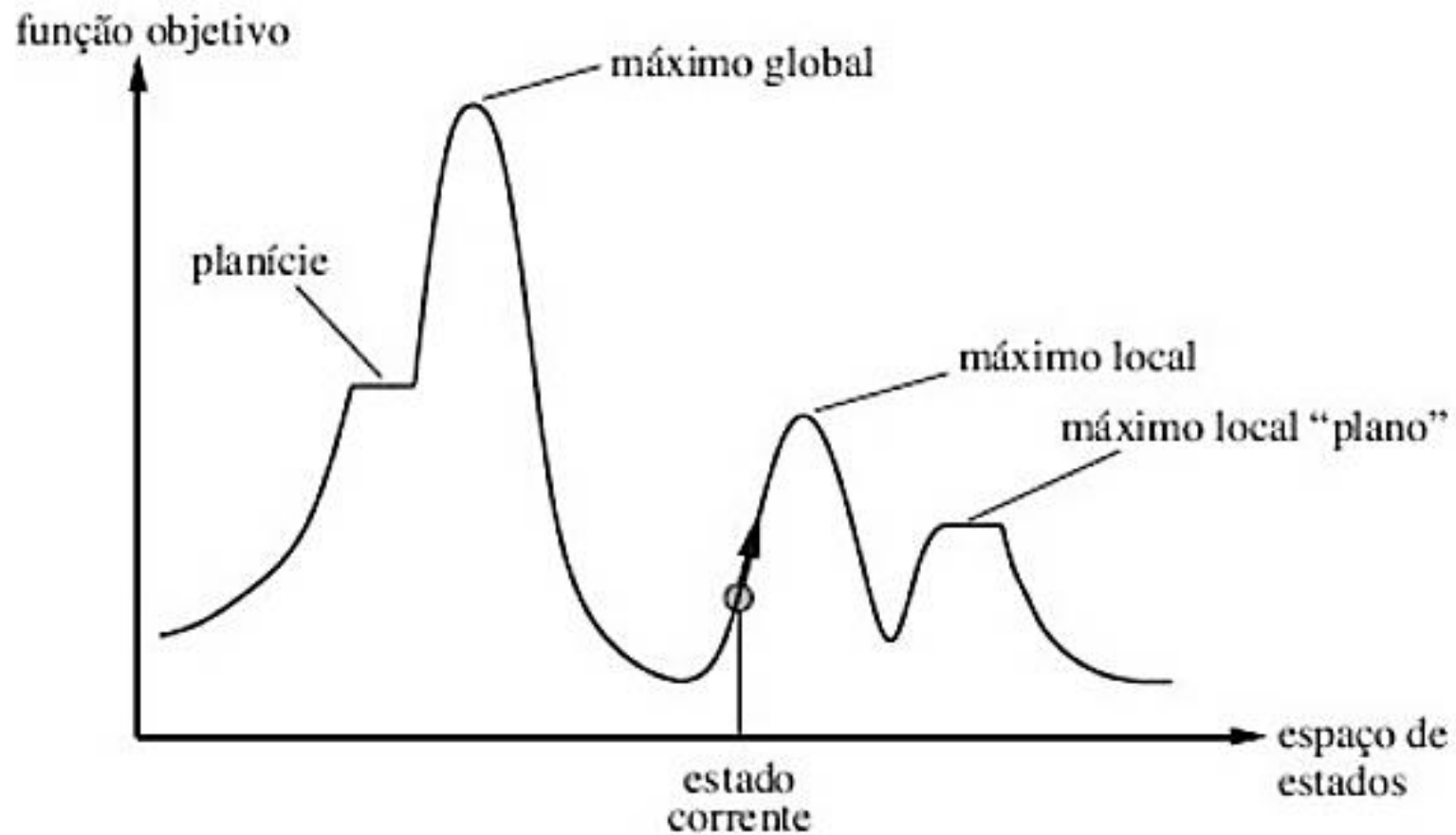
Têmpera simulada

$$p(\Delta E) = e^{-\Delta E / T}$$

- Probabilidade também decresce à medida que a “temperatura” T se reduz;
- Movimentos “ruins” têm maior probabilidade de serem permitidos no início, quando T estiver alto. Tornam-se mais improváveis conforme T diminui;
- Se o escalonamento diminuir T com lentidão suficiente, o algoritmo encontrará um valor ótimo global com probabilidade próxima de 1.

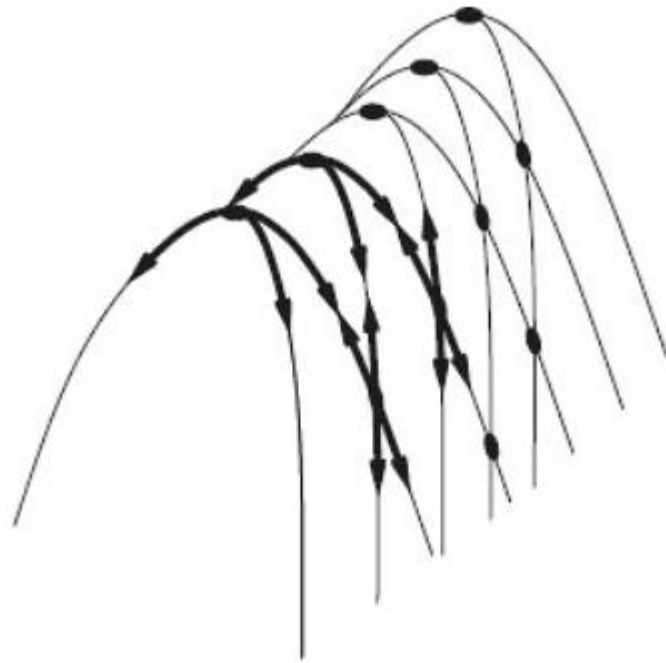


Têmpera simulada





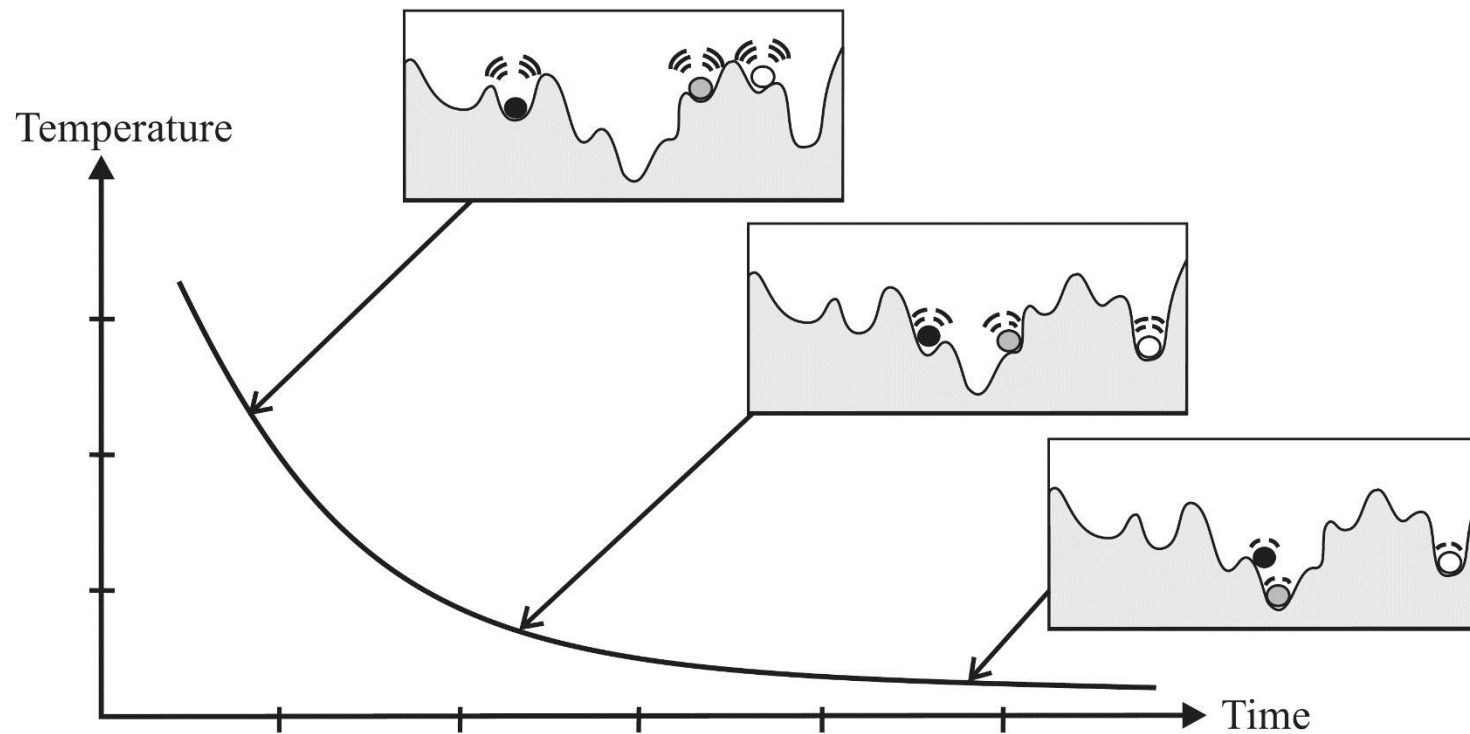
Têmpera simulada



Cordilheiras

- Sequência de máximos locais que torna muito difícil a navegação para algoritmos gulosos;
- A partir de cada máximo local, todas as ações disponíveis apontam encosta abaixo

Têmpera simulada





Têmpera simulada

- A temperatura é elevada e a probabilidade de se aceitar soluções de piora é maior no início do processo;
- Possibilidades de pioras são aceitas para escapar de mínimos (ou máximo) locais;
- Probabilidade de se aceitar uma solução pior depende da temperatura (T);
 - Quanto maior T , maior a probabilidade.



Têmpera simulada

- Número de iterações máximo em uma temperatura X deve ser calculada de acordo com a dimensão do problema;
- Congelamento do sistema: quando se atingir um valor ($T = 0,001$, por exemplo) ou quando a taxa de aceitação atingir um limite;
- Obtenção dos parâmetros mais adequados podem ser obtidos por meio de experimentação.

Têmpera simulada

```
procedimento SA ( $f(.)$ ,  $N(.)$ ,  $\alpha$ ,  $SA_{max}$ ,  $T_0$ ,  $s$ )  
   $s^* \leftarrow s$       {Melhor solução obtida até então}  
  IterT  $\leftarrow 0$    {Número de iterações na temperatura  $T$ }  
   $T \leftarrow T_0$      {temperatura corrente}  
  enquanto ( $T > 0.0001$ )  
    enquanto (IterT  $< SA_{max}$ ) faça  
      IterT  $\leftarrow$  IterT + 1  
      Gerar um vizinho ( $s'$ ) aleatoriamente na vizinhança  $N^k(s)$   
       $\Delta = f(s') - f(s)$   
      se ( $\Delta < 0$ ) então  
         $s \leftarrow s'$   
        se ( $f(s') < f(s^*)$ ) então  $s^* \leftarrow s'$   
      senão  
        Tome  $x \in [0,1]$   
        se ( $x < e^{-\Delta/T}$ ) então  
           $s = s'$   
      fim-se  
    fim-enquanto  
     $T = T \times \alpha$   
    IterT = 0  
  fim-enquanto  
  retorne  $s^*$   
fim-procedimento
```



Têmpera simulada

- T Inicial deve ser alto o suficiente para permitir movimentos livres entre soluções vizinhas
 - Pode ser obtida por simulação (fixando uma taxa de mínima de movimentos aceitos, por exemplo)
- Equilíbrio térmico deve ser aproximado a cada T;
- Teoricamente O T final deve ser zero
 - Na prática basta chegar a uma temperatura próxima a zero – implementação computacional



Referências

- RUSSELL, Stuart; NORVIG, Peter; INTELLIGENCE, Artificial. A modern approach. **Artificial Intelligence. Prentice-Hall, Egnlewood Cliffs**, v. 25, p. 27, 1995
- KIRKPATRICK, GELATT; M. P. VECCHI; Optimization by Simulated Annealing, Science
- FERREIRA, V. H. Introdução à Inteligência Computacional – UFF
- POZO, A. Implementações do Simulated Annealing para o problema do Caixeiro Viajante Simétrico