INTELIGÊNCIA ARTIFICIAL

ESTRATÉGIAS DE BUSCA LOCAL

Prof. Chauã Queirolo

1

SUMÁRIO

2

Introdução

- Um problema pode ser definido por 5 componentes
  - Estado inicial
  - Ações
  - Modelo de transição
  - Teste de objetivo
  - Custo do caminho

3

Introdução

Solução

- Sequência de ações que levam do **estado inicial** para o **estado objetivo**

Solução ótima

- Solução com o menor **custo de caminho**

4

Busca local

5

Busca local

- Em muitos problemas de otimização o **caminho** para o objetivo é irrelevante
  - Queremos apenas encontrar o **estado objetivo**
  - **Não importa** a sequência de **ações**

6

Busca local

Algoritmos de busca local

- Mantêm apenas o estado **atual**
- Sem a necessidade de manter a **árvore de busca**

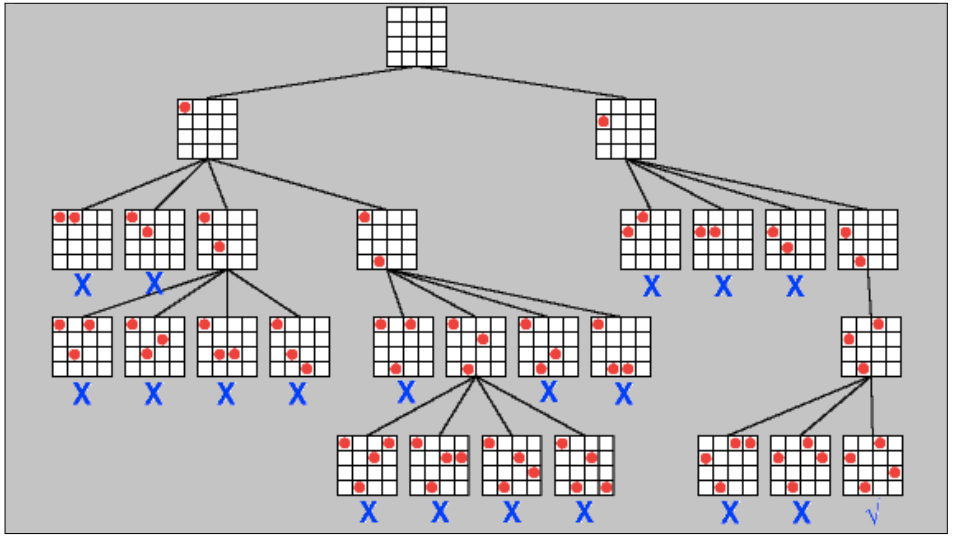
7

Busca local

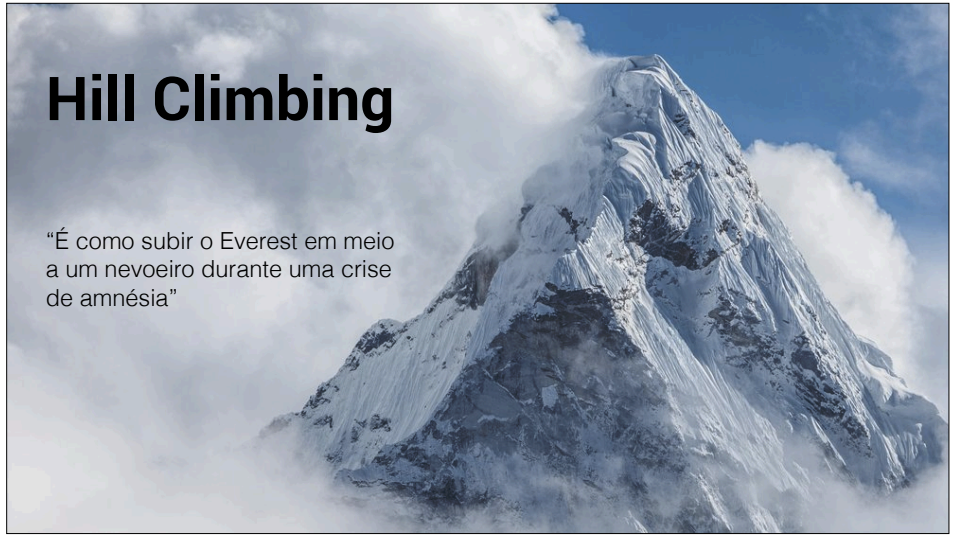
Problema das 8 rainhas

Colocar n rainhas em um tabuleiro n × n, sendo que cada linha coluna ou diagonal pode ter apenas uma rainha

8



9



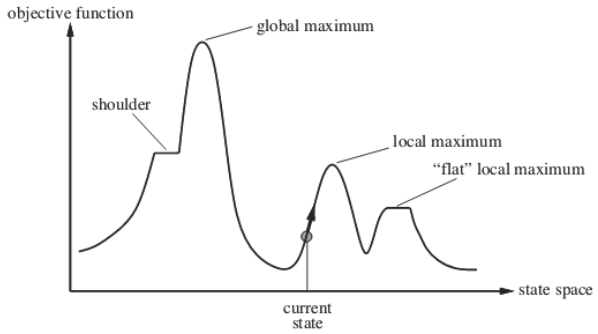
10

Hill Climbing

1: i = initial solution
2: **While** $f(s) \leq f(i)$ $s \in$ Neighbours (i) **do**
3: Generates an $s \in$ Neighbours (i);
4: If fitness (s) > $fitness$ (i) **then**
5: Replace s with the i;
6: **End If**

11

Hill Climbing



12

Hill Climbing

- Elevação
  - **Função objetivo:** queremos encontrar o máximo global
  - **Custo:** queremos encontrar o mínimo global
- O algoritmo consiste em uma repetição que percorre o espaço de estados no sentido do **valor crescente** (ou decrescente)
- **Termina** quando encontra um **pico** (ou **vale**) em que nenhuma vizinho tem valor mais alto

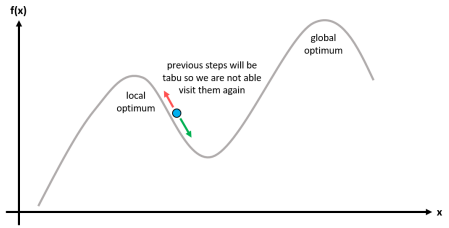
Hill Climbing

Variantes

- Steepest-Ascent Hill climbing
  - Examina todos os vizinhos e escolhe o melhor
- Stochastic Hill climbing
  - Seleciona k vizinhos aleatórios e escolhe o melhor
- Hill Climbing Random Restart
  - Inicializa o Hill Climbing em diferentes pontos do espaço de busca

Busca Tabu

- Utiliza uma memória auxiliar com estados já visitados → **Lista tabu**
- Estados na lista tabu não são visitados de novo



Busca Tabu

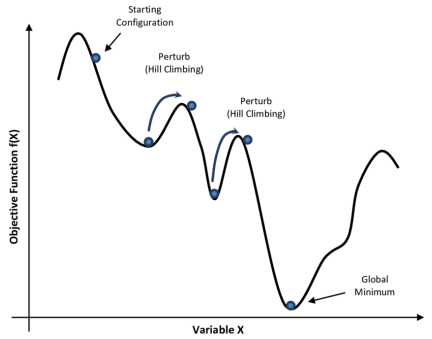
MAXITER : the maximum number of iterations
$x' \leftarrow$ produce an initial solution x
initialize tabu list T

1. **for** i = 1 **to** MAXITER **do**
2. identify Neighborhood set N
3. identify Candidate set $C = N - T + AC$
4. find the best x from C
5. **if** $f(x) > f(x')$ **then**
6. $x' \leftarrow x$
7. **end if**
8. update T with FIFO policy
9. **End for**

Simulated annealing

- Analogia com o processo de arrefecimento dos metais
  - Um metal é aquecido e resfriado lentamente
  - A medida que resfria, as moléculas se organizam em uma estrutura mais sólida
- A escolha dos vizinhos é definida de acordo com a temperatura
  - Temperatura alta: maior chance de selecionar soluções candidatas piores
  - Temperatura baixa: sistema estabiliza e menor chance de escolher soluções piores

Simulated annealing



Simulated annealing

```
1  Construct the initial solution S
2  S* = S, T = T_0, T_b = T_0
3  while time limit is not exceeded
4      for k = 1 to Len
5          Select a neighborhood structure NS randomly
6          Generate a feasible solution S' from S with NS
7          if cost(S') < cost(S)
8              S = S'
9          else
10             Set S = S' with probability p, where p = exp(-(cost(S') - cost(S))/T)
11         if S' is better than S*
12             S* = S', T_b = T
13     T = α * T
14     if T < 0.01
15         T_b = 2 * T_b, T = min{T_b, T_max}
16 return S*
```


REFERÊNCIAS BIBLIOGRÁFICAS

– S. J. Russell & P. Norvig. **Artificial Intelligence: A Modern Approach.** Prentice Hall, 3rd edition, 2010.