

Pesquisa Local

Trepa-colinas

Recristalização Simulada

Pesquisa Local

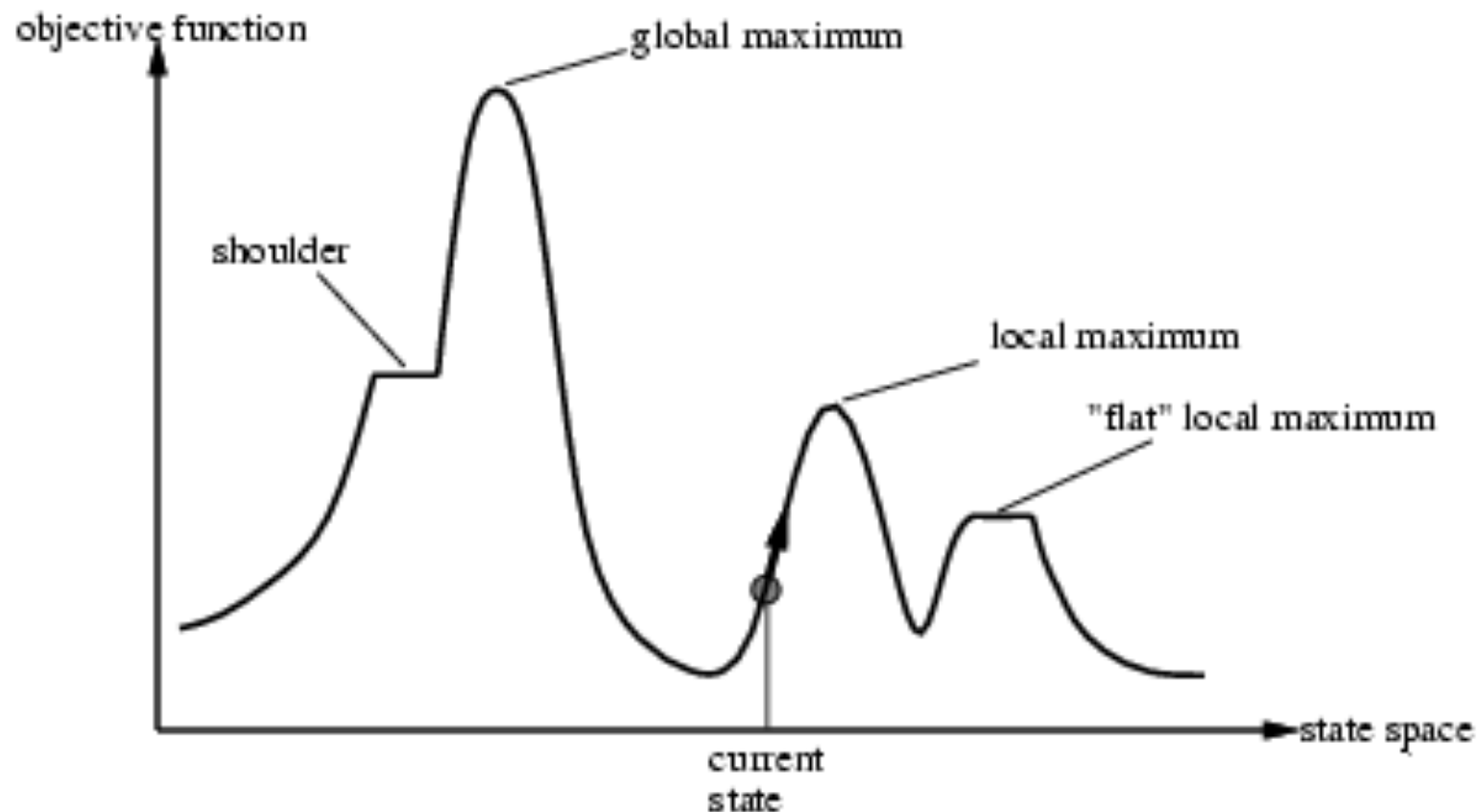
- **Representação:**
 - Como representar uma solução para o problema?
- **Função de avaliação:**
 - Atribui a qualidade à solução
 - Identificar se o problema é de minimização ou maximização
 - Há a possibilidade de surgirem soluções inválidas?
 - Penalizar
 - Reparar
- **Vizinhança:**
 - Cria uma nova solução a partir da solução atual

Trepa-Colinas

```
Trepa-Colinas(solucao, k)
    custo = avalia(solucao)
    Repete k iteracoes
        nova_sol = gera_vizinho (solucao)
        novo_custo = avalia(nova_sol)
        se novo_custo > custo (maximização) então
            solucao = nova_sol
            custo = novo_custo
        fim_se
    Fim_repete
```

Trepa-Colinas

- Limitações: ótimos locais



Recristalização Simulada

- Ideia: fugir aos ótimos locais permitindo a aceitação de soluções piores de forma “controlada”

Recristalização_simulada(solucao, k)

t = TMAX

itera = 0

custo = avalia(solucao)

ENQUANTO t > TMIN

Repete k vezes

nova_sol = gera_vizinho (solucao)

novo_custo = avalia(nova_sol)

SE novo_custo > custo (maximização) ENTÃO

solucao = nova_sol

custo = novo_custo

SENÃO //aceita solução pior

SE rand_01() <= exp(**(novo_custo-custo)/t**) ENTÃO

solucao = nova_sol

custo = novo_custo

FIM_SE

FIM_SE

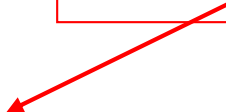
Fim_Repete

itera = itera + 1

t = t * fator_arrefecimento

FIM_ENQUANTO

Para maximização!
Se for minimização substituir por :
(custo – novo_custo)



Recristalização Simulada

- Se a temperature decrescer lentamente e se atingir um valor muito próximo de zero, com o tempo necessário, este algoritmo encontra a solução ótima do problema.