

Conferencia 10
Métodos Locais

Objetivo

- Caracterizar os métodos de busca local

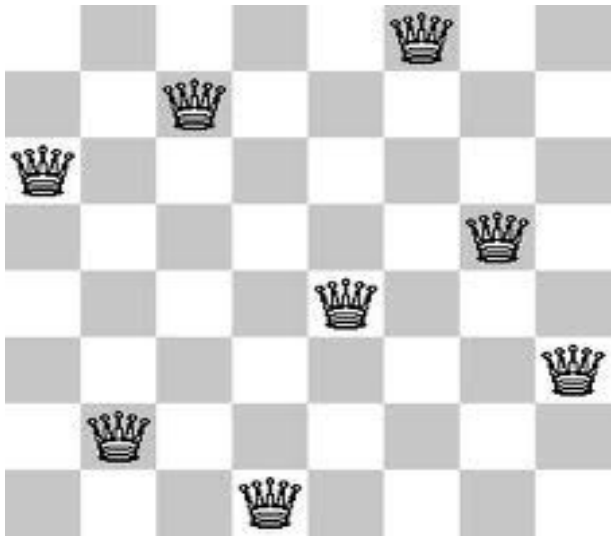
Bibliografía

- Patrick Henry Winston, Inteligencia Artificial, 3ra edición, 1992.
- AI. A Modern Approach Russell Norvig 3rd ed. 2010.
- Elaine Rich & Kevin Knight, Inteligencia Artificial, 2da edición, 1994.
- Rafael Bello, Métodos de solución de problemas, 1998.

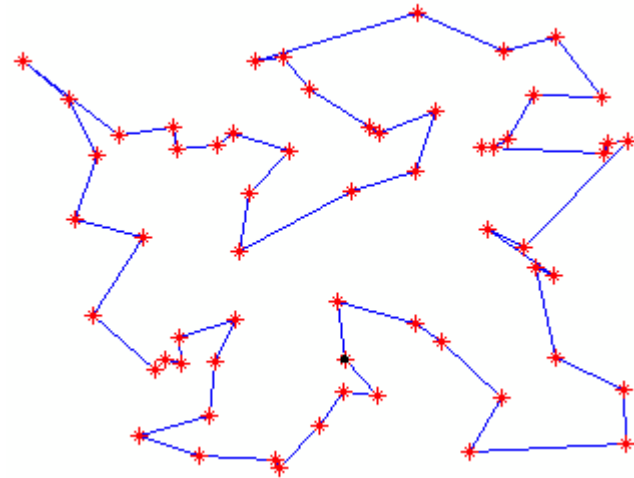
Resumo

- Hill Climbing
- Recozimento Simulado

Introdução



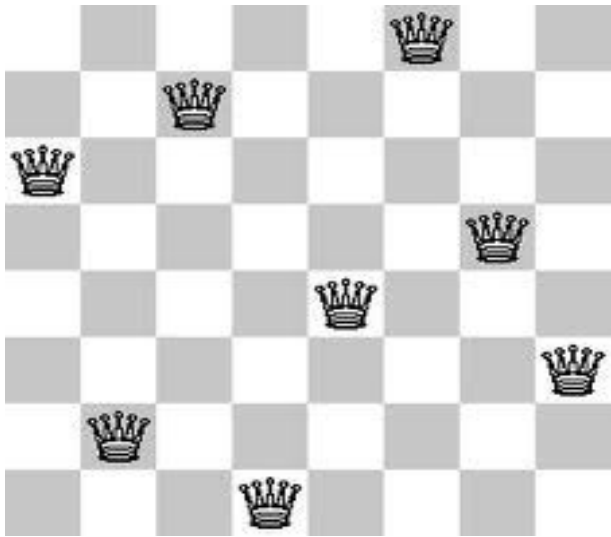
8 rainhas



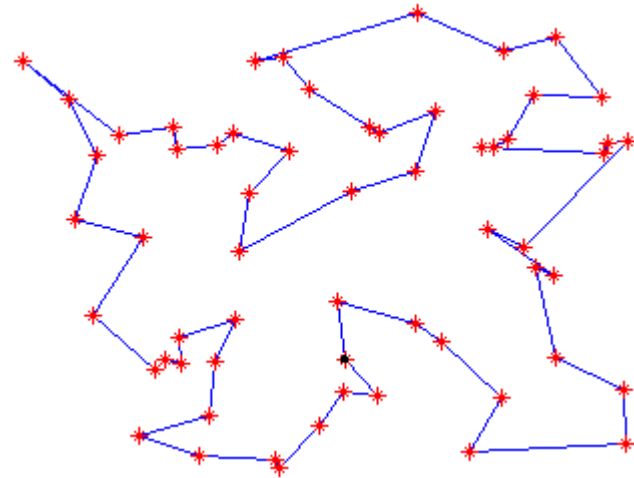
Vendedor ambulante

A maneira de alcançar a solução é interessante?

Introdução



8 rainhas



Vendedor ambulante

A maneira de alcançar a solução é interessante?

IRRELEVANTE

Introdução

Áreas de aplicação:

- problemas de otimização
- design de circuitos integrados
- programação automática
- problemas de agendamento
- otimização em redes de telecomunicações

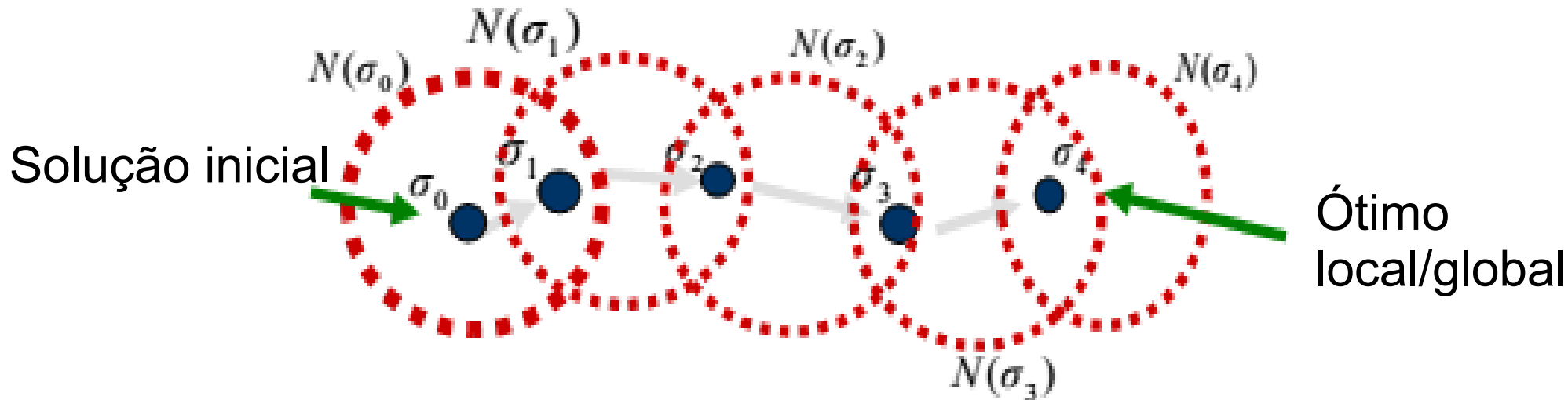
Introdução

O termo **LOCAL** é freqüentemente usado em estudos teóricos e práticos de metaheurísticas de busca. Está associado ao uso de estruturas ambientais, refletindo o conceito de **proximidade** ou **vizinhança** entre soluções alternativas do problema.

Todas as soluções incluídas no **ambiente de solução atual**, que é delimitado por um operador de geração de solução, são chamadas de **soluções vizinhas**.

Introdução

Assim, os algoritmos baseados nesta estratégia fazem um **estudo local do espaço de busca**, pois analisam o ambiente da solução atual para decidir como continuar o caminho da pesquisa.



Introdução

Uma **busca local** é um processo que, dada a solução atual em que se encontra o percurso, seleciona de forma iterativa uma solução de seu entorno para continuar a busca.

Estrutura de Entorno

Basta desenhando a estrutura de entorno para obter um modelo genérico de algoritmo de busca.

Descrição

- Se fixa uma codificação para as soluções.
- Define-se um operador de geração de vizinho e, em consequência, se fixa uma estrutura de entorno para as mesmas.
- Escolhe-se uma solução do entorno da solução atual até que se satisfaça o critério de parada.

Estrutura de Entorno

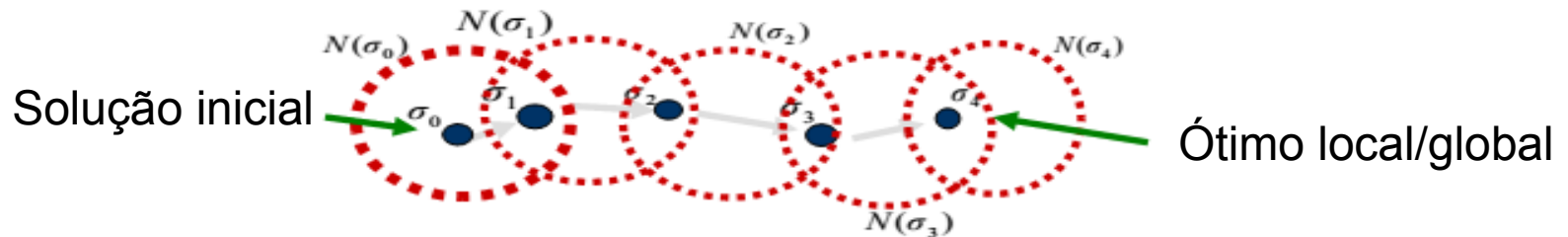
$$S \rightarrow S', S' \in E(S) \text{ (também notado } N(S)).$$

Elementos básicos:

- Processo de eleição da solução inicial.
- Processo de seleção de solução / geração de uma solução vizinha:

$$S \rightarrow S', S' \in E(S) \text{ (também denotado } N(S))$$

- Processo de aceitação de solução vizinha como solução atual.



Procedimento Busca por Entornos

Procedimento Busca por Entornos

Início

GERA(Solução Inicial)

Solução Atual <- Solução Inicial;

Melhor Solução <- Solução Atual;

Repetir

Solução Vizinha <- GERA_VIZINHO(Solução Atual);

Se Aceitar(Solução Vizinha)

então Solução Atual \u2190 Solución Vizinha;

Se Objetivo(Solução Atual) é melhor que Objetivo(Melhor Solução)

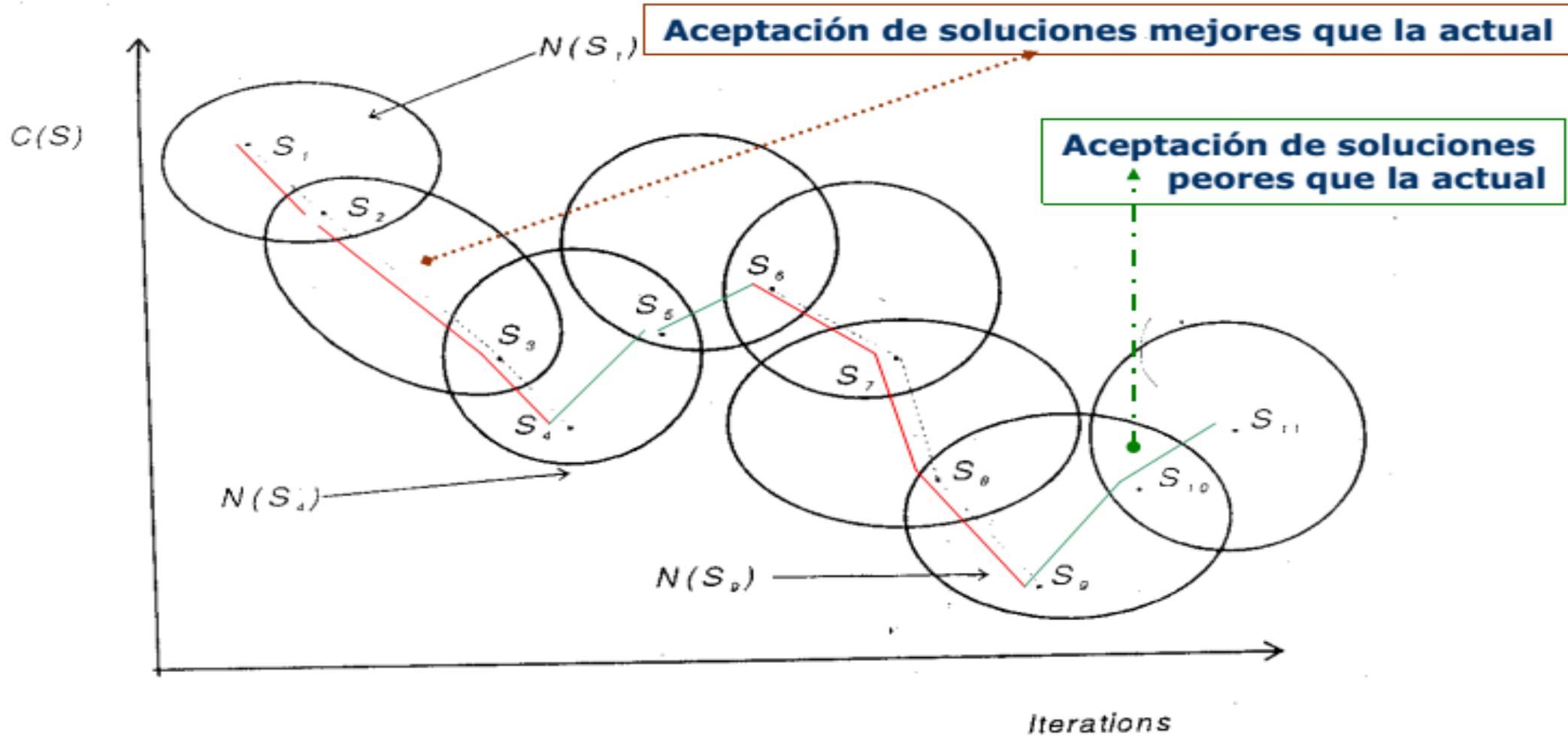
então Melhor Solução \u2190 Solución Atual;

Até(Critério de parada);

DEVOLVER (Melhor Solução);

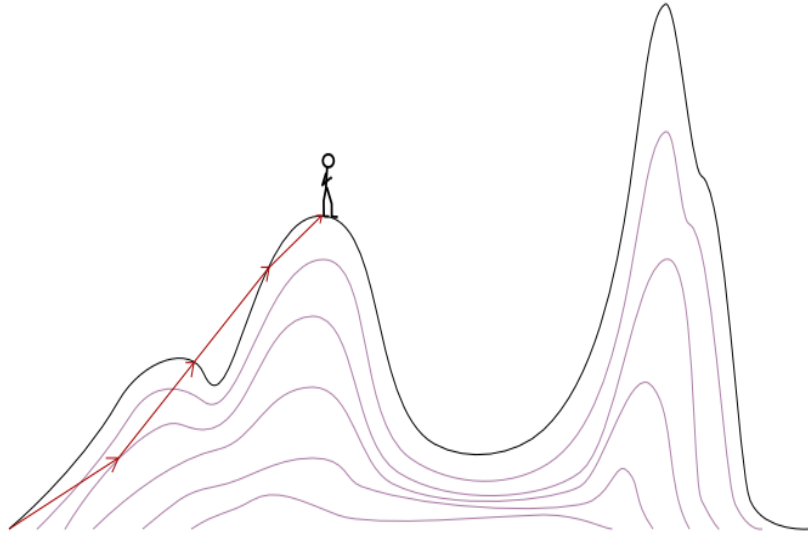
Fim

Introdução



Hill climbing

Toma seu nome da semelhança que tem com um alpinista quem deseja alcançar rapidamente o pico de uma montanha, este seleciona a direção de ascensão maior a partir da posição atual.



Hill climbing

É um ciclo que se move na direção de máxima ascensão ou descida, em dependência de se se esteja maximizando ou minimizando a função objetivo.

Com este método a estratégia é repetidamente expandir um nó, inspecionar seus sucessores recém gerados, e selecionar e expandir o melhor entre os sucessores sem manter referências aos pais.

Hill climbing

O algoritmo não mantém uma árvore de busca, de tal forma que o nodo atual solo precisa armazenar o estado e sua avaliação da função objetivo.

Este método é uma **variação** da **primeiro busca em profundidade** na qual se usa a função heurística para estimar a distância entre o nodo atual e o nodo objetivo. Por exemplo, a distância aérea no problema do mapa.

Hill climbing

- P1. Formar uma pilha de um elemento consistente do nodo raiz.
- P2. Iterar de (a) a (c) até que a pilha este vazia.
 - A. Checar se o elemento do Top da pilha é uma solução.
 - B. SAIR com ÊXITO se o elemento checado é solução.
 - C. Se o primeiro elemento não é solução ordenar os descendentes do mesmo segundo a distância restante ao objetivo e logo acrescentar o filho ao Top de pilha.
- P3. Sair com Falha.

Hill climbing

Algumas observações sobre o método:

- Quando se chega a um nodo morto não há forma de fazer retrocesso (salvo gerar outro nodo raiz)
- É usado quando existe uma boa função heurística
- A estratégia é chamada **irrevogável** porque o processo não nos permite virar a atenção para alternativas previamente suspensas

Hill climbing

Algumas observações sobre o método:

- A estratégia é também útil em problemas onde os operadores possuem um certo tipo de independências chamada **comutatividade**, uma condição na qual a aplicação de um operador não afeta a aplicabilidade de outros operadores.

Steepest-Ascent Hill climbing

P1. Avaliar o estado inicial. Se for objetivo SAIR com ÊXITO

P2. Iterar até que uma solução se encontre ou até que uma reiteração completa não produza mudança do estado atual:

a. Seja SUCC um estado tal que qualquer possível sucessor do estado atual será melhor que SUCC.

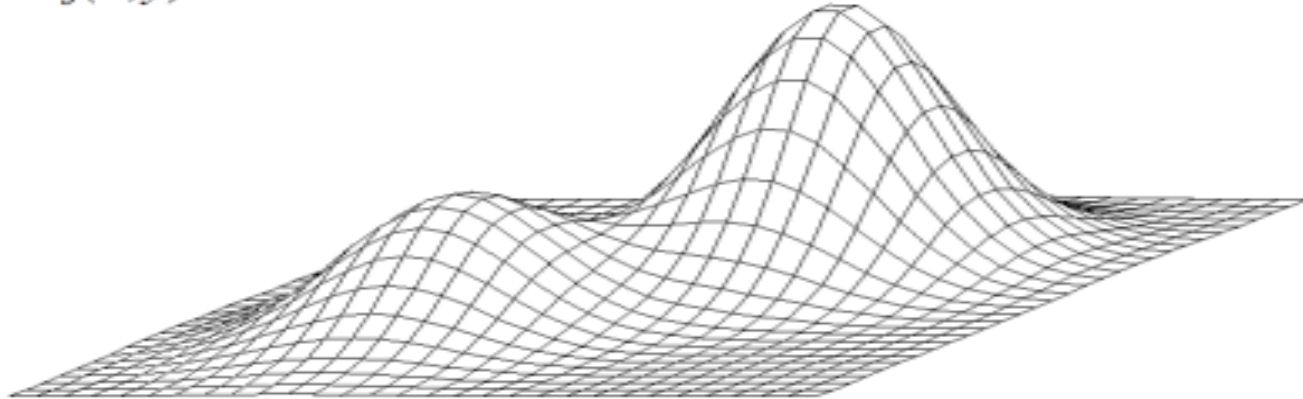
Steepest-Ascent Hill climbing

- b. Para cada operador aplicável ao estado atual fazer:
 - i. Aplicar o operador e gerar um novo estado.
 - ii. Avaliar o novo estado. Se for objetivo SAIR com ÊXITO sina compará-lo com o SUCC. Se for melhor atribuir SUCC a esse estado.
- c. Se SUCC for melhor que o estado atual, então fazer o estado atual ao SUCC.

Hill climbing. Condições de falha

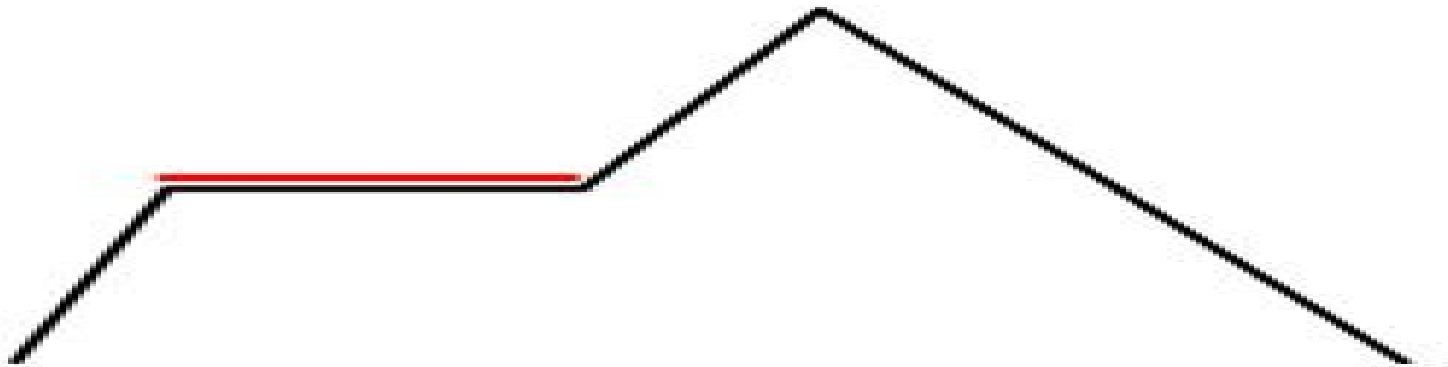
a) Um **máximo local**: é um estado que é melhor que todos seus vizinhos mas não é melhor que alguns outros estados mais longínquos. Em um máximo local todos os movimentos parecem piorar as coisas. Freqüentemente surgem perto de uma solução.

$$f(x,y) = e^{-(x^2+y^2)} + 2e^{-((x-1.7)^2+(y-1.7)^2)}$$



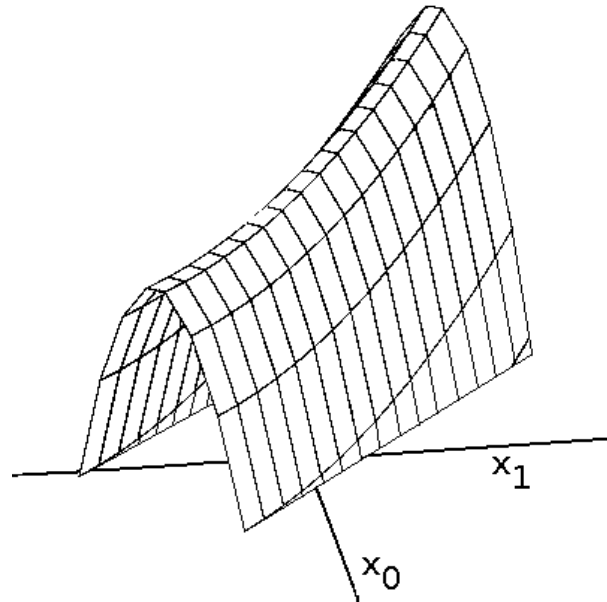
Hill climbing. Condições de falha

b) Uma **meseta (plateau)** é uma área plana no espaço de busca na qual todo o conjunto de estados vizinhos tem o mesmo valor. Em uma meseta não é possível determinar a melhor direção para nos mover fazendo comparações locais.



Hill climbing. Condições de falha

- c) Uma **crista (ridge)** é uma área do espaço de busca que é mais alta que áreas adjacentes mas não pode atravessar-se mediante movimentos elementares em qualquer direção.



Simulated annealing

O **Esfriamento** ou **Recozimento Simulado** é um algoritmo de busca por entornos com um critério probabilista de aceitação de soluções apoiado em Termodinâmica.

Simulated annealing. Fundamentos

- Um modo de evitar que a busca local finalize em ótimos locais, feito que está acostumado a ocorrer com os algoritmos tradicionais de busca local é **permitir que alguns movimentos sejam para soluções piores**.
- Mas se a busca está avançando realmente para uma boa solução, estes movimentos "de escapamento de ótimos locais" devem realizar-se de um modo **controlada**.
- Isto se realiza controlando a frequência dos movimentos de escapamento mediante uma função de probabilidade que fará diminuir a probabilidade destes movimentos para soluções piores conforme avança a busca.

Modelo de Metropolis

- Apóia-se no trabalho de Metrópolis (1953) no campo da termodinâmica estatística.
- Basicamente modelo o processo de esfriamento simulando as mudanças energéticas em um sistema de partículas conforme decresce a temperatura, até que converge a um estado estável (congelado).
- As leis da termodinâmica dizem que a uma temperatura t a probabilidade de um incremento energético de magnitude δE se pode aproximar por

$$P[\delta E] = e^{-\delta E / kt}$$

sendo k uma constante física denominada Boltzmann.

Analogias: Termodinâmica - Otimização

Simulação termodinâmica

- Estado do Sistema
- Energia
- Mudança de status
- Temperatura
- Estado congelado

Otimização combinatória

- Soluções viáveis
- Custo
- Soluções na vizinhança
- Parâmetro de controle
- Solução heurística

Simulated annealing. Características gerais

En cualquier momento pueden ocurrir movimientos de ascenso, aunque con menos probabilidad y magnitud al final del proceso hasta que finalmente el proceso converge a un mínimo.

El proceso es gobernado por el parámetro T . El plan de cómo variar T se denomina **esquema de recocido**.

Un enfriamiento demasiado rápido puede llevar a regiones estables de alta energía (mínimos locales en lugar de globales). Un enfriamiento lento propicia alcanzar con más seguridad una estructura uniforme (mínimo global), pero si es demasiado lento se pierde tiempo.

Simulated annealing. Algoritmo

P1. Avaliar o estado inicial. Se for objetivo SAIR com ÊXITO, sina fazê-lo estado atual.

P2. Atribuir ao BEST-SO-FAR o estado atual.

P3. Inicializar T segundo o esquema de recocido.

P4. Iterar até encontrar uma solução ou até que não haja novos operadores aplicáveis ao estado atual.

a) Selecionar um operador não usado e aplicá-lo ao estado atual para produzir um novo estado.

Simulated annealing. Algoritmo

b) Avaliar o novo estado. Calcular.

$\Delta E = (\text{valor do estado atual}) - (\text{valor do novo estado})$

i. Se o novo estado é objetivo, SAIR com ÊXITO

ii. Se não ser objetivo mas é melhor que o atual, fazê-lo atual. Atribui-lo ao BEST-SOU-FAR

iii. Se não ser melhor então calcular $P' = e^{-\Delta E/T}$

iv. Se não ser melhor então calcular

Gerar um número aleatório a na intervalo $[0,1]$.

Se $a < p'$ então fazer o movimento.

Simulated annealing. Algoritmo

c) Revisar T de acordo ao esquema de percurso

P5. Retornar BEST-SO-FAR como a resposta

O esquema de percurso tem três componentes. A primeira é o valor inicial para o T. O segundo o critério para decidir quando reduzir a T. O terceiro é a quantidade para a qual T deve ser reduzida cada vez que seja trocada.

Parâmetros e componentes

- Representação
- Geração da solução inicial
- Mecanismo de transição de soluções
- Seqüência de esfriamento

Representación. Ejemplos

- **Vetor ordenado de números inteiros.**

Exemplo: Problema do Viajante de Comércio

(7 6 3 1 5 4 2 8)

- **Vetor binário.**

Exemplo: Problema da mochila

(0 1 0 0 1 1 1 0)

- **Vetor de Números Reais**

Exemplo: Problemas de Otimização com Parâmetros Contínuos

(2 7 3 5 4 6 27)

Geração da solução inicial

- Uso de técnicas eficientes para obtê-lo
- Uso de conhecimento especializado

Exemplo: solução gerada por um algoritmo greedy

Mecanismo de Transição de Soluções

1. Geração de uma nova solução

- Definição do conjunto de vizinhos
- Seleção de um elemento de dito conjunto

2. Cálculo da diferença de custos entre a solução atual e a vizinha

3. Aplicação do critério de aceitação

Seqüência de resfriamento

1. Valor inicial do parâmetro de controle

Não parece apropriado considerar valores fixos independentemente do problema.

Proposta:
$$T_0 = (\mu / -\ln(\varphi)) * C(S_0)$$

Tanto para uma probabilidade φ que uma solução é um μ por um pior do que a solução inicial S_0

Seqüência de resfriamento

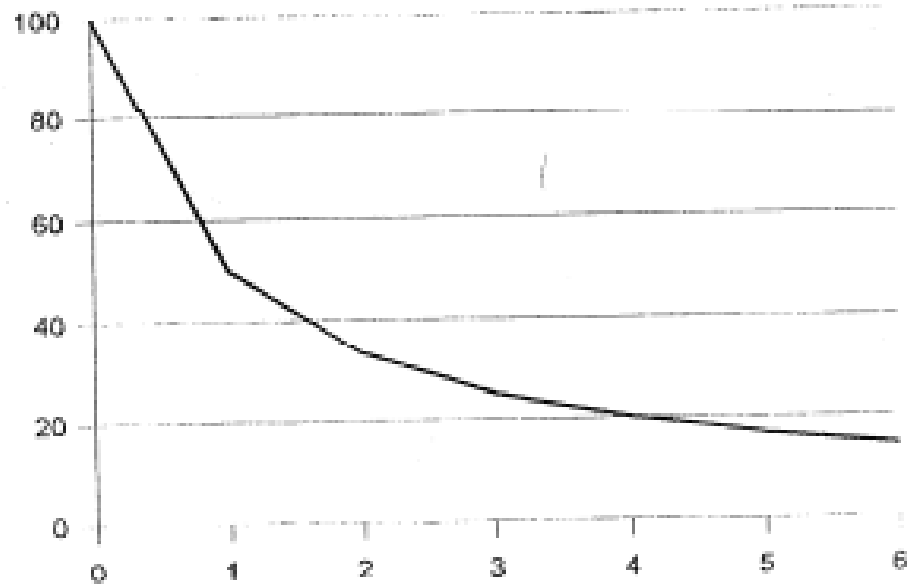
2. Mecanismo de enfriamento

- Esfriamento apoiado em sucessivas temperaturas descendentes fixadas pelo usuário
- Esfriamento com descida constante de temperatura
- Descida exponencial: $T_{k+1} = \alpha * T_k$ k é o número da reiteração atual, α uma constante próxima a 1 (usualmente, $\alpha \in [0.8, 0.99]$)
- Critério do Boltzmann: $T_k = T_0 / (1 + \log(k))$
- Esquema do Cauchy: $T_k = T_0 / (1 + k)$

Seqüência de resfriamento

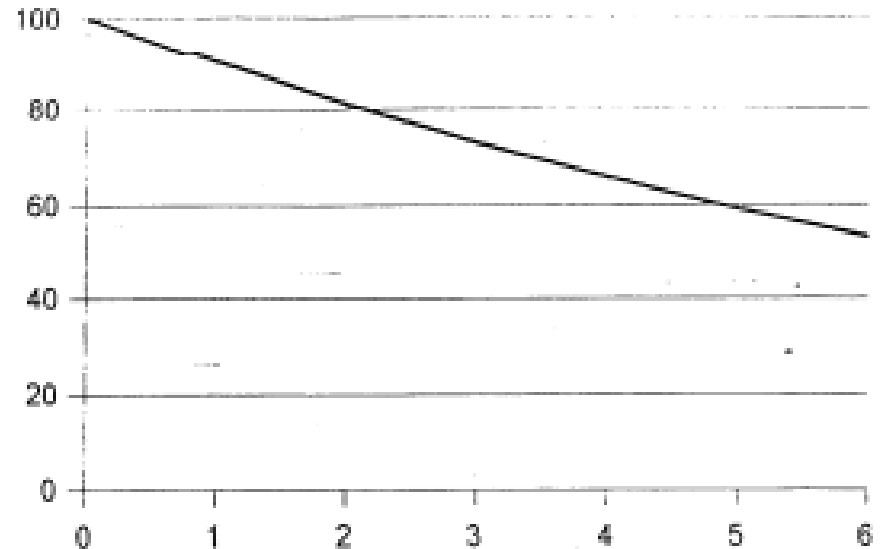
2. Mecanismo de enfriamiento

Esquema de enfriamiento de Cauchy



Esquema de enfriamiento proporcional

Parametro de proporcionalidad : 0.9



Seqüência de resfriamento

3. Velocidade de resfriamento

4. Condição de parada

Em teoria, o algoritmo deve terminar quando $T = 0$. Na prática, ele pára quando:

- quando T alcança ou está abaixo de um valor final T_f , pré-definido ou
- depois de um número fixo de iterações
- Uma boa opção é parar quando nenhum vizinho foi aceito a partir dos gerados na iteração atual.

Simulated annealing

INPUT (T_0, α, L, T_f)

$T \leftarrow T_0$

Valor inicial del parámetro de control

$S_{act} \leftarrow \text{Genera_solucion_inicial}$

WHILE $T \geq T_f$ DO

Condición de Parada

BEGIN

FOR cont $\leftarrow 1$ TO $L(T)$ DO

Velocidad de enfriamiento $L(T)$

BEGIN

$S_{cand} \leftarrow \text{Selecciona_solucion_N}(S_{act})$

Generación de una nueva solución

$\delta \leftarrow \text{coste}(S_{cand}) - \text{coste}(S_{act})$

Cálculo de la diferencia de costos

IF $(U(0,1) < e^{(-\delta/T)})$ OR

$(\delta < 0)$ THEN $S_{act} \leftarrow S_{cand}$

Aplicación del criterio de aceptación

END

$T \leftarrow \alpha(T)$

Mecanismo de enfriamiento

END

{Escribe como solución, la mejor de las S_{act} visitadas}

CONCLUSÕES