Resolução de Problemas por Buscas

Danilo Sipoli Sanches

Departamento Acadêmico de Computação
Universidade Tecnológica Federal do Paraná
Cornélio Procópio

Resolução de Problemas por Buscas - Clássicas

- **Interesse:** buscar sistematicamente, a partir de um estado inicial, uma sequência de ações que levem à meta
- **Solução:** caminho ao estado-objetivo
- Algoritmos de busca exploram o espaço de estados
  - mantêm um ou mais caminhos na memória
  - registram alternativas exploradas e não exploradas

Resolução de Problemas por Buscas - Clássicas

- Problemas de otimização → **caminho** ao objetivo é **irrelevante**
- **Solução** = estado objetivo

Exemplos

- projeto de circuitos integrados
- layout de instalações industriais
- escalonamento de trabalhos
- otimização de redes
- gerenciamento de salas de aula
- problema da mochila (*knapasack*)
- entre outros ...

Descrição do estado contém toda informação relevante para a solução → Caminho até a meta não importa

Além da Busca Clássica

Busca Local ou de melhoria iterativa

Operam em um único estado e move-se para a vizinhança deste estado

Ideia

Começar com o estado inicial (configuração completa, solução aceitável), e melhorá-lo iterativamente

Vantagens

- utiliza pouca memória (usualmente uma quantidade constante)
- frequentemente encontra boas soluções em espaço de estados muito grandes ou mesmo infinitos, nos quais estratégias sistemáticas são inadequadas ou inviáveis

Busca Local

- Úteis para resolver **problemas de otimização**
  - buscar por estados que atendam a uma **função objetivo**

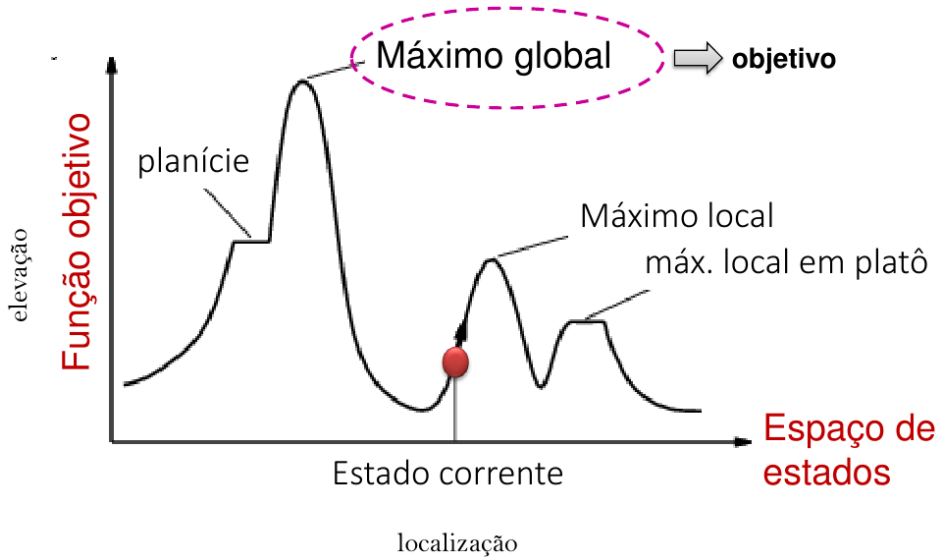
Objetivo

Encontrar o estado que:

- maximize a função objetivo ou
- minimize o custo de uma heurística h(n)

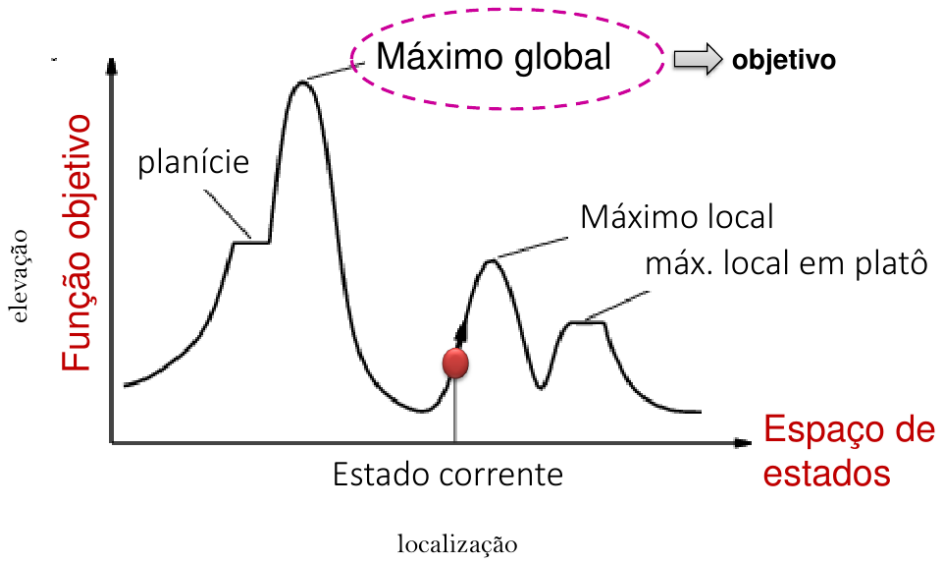
Busca Local

- Estados podem ser representados sobre uma superfície
  - altura de qualquer ponto na superfície corresponde à função de avaliação f(n) do estado naquele ponto
- Algoritmo move pela superfície em busca de pontos mais altos/mais baixos (**solução ótima**)



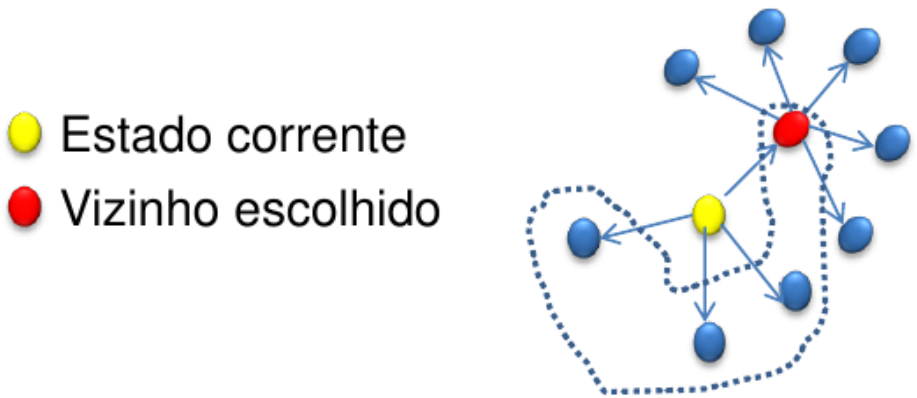
Busca Local

- **Completo:** consegue atingir um estado objetivo, caso exista
- **Ótimo:** consegue encontrar o mínimo/máximo global



Busca Local

- Algoritmos guardam apenas o estado corrente, e não veem além dos vizinhos imediatos do estado
- Muitas vezes são os melhores métodos para tratar problemas reais muito complexos

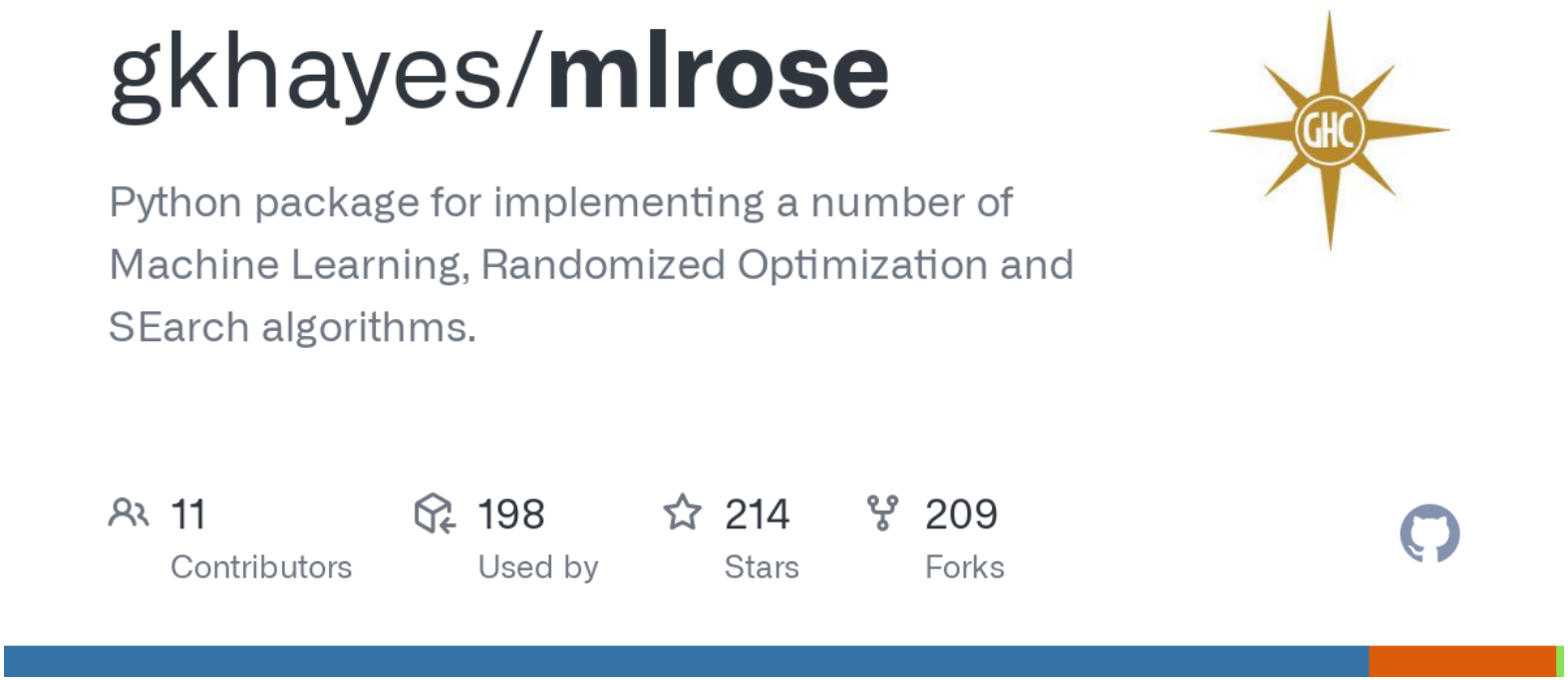


Busca Local

- Principais algoritmos
  - Hill-Climbing
  - Simulated Annealing
  - Genetic Algorithms

Framework: MIrose

- Hayes, G. (2019). mlrose: Machine Learning, Randomized Optimization and SEarch package for Python. https://github.com/gkhayes/mlrose



Framework: EasyGA

https://pypi.org/project/EasyGA/1.0.1/

Framework: DEAP

https://deap.readthedocs.io/en/master/

DISTRIBUTED
EVOLUTIONARY
ALGORITHMS IN
PYTHON


Obrigado!