

Inteligência Artificial

Busca Cega ou Exaustiva

Prof. Dr^a. Andreza Sartori
asartori@furb.br

Documentos Consultados/Recomendados

- KLEIN, Dan; ABBEEL, Pieter. **Intro to AI**. UC Berkeley. Disponível em: <http://ai.berkeley.edu>.
- FARIA, Fabio Augusto. **Inteligência Artificial**. Universidade Federal de São Paulo, 2017.
- LIMA, Edirlei Soares. **Inteligência Artificial**. PUC-Rio, 2015.
- RUSSELL, Stuart J. (Stuart Jonathan); NORVIG, Peter. **Inteligência artificial**. Rio de Janeiro: Campus, 2013. 1021 p, il.
- VIERIU, Radu-Laurențiu. **Artificial Intelligence**. Università degli Studi di Trento, 2016.

Conteúdo Programático

Unidade 1: Fundamentos de Inteligência Artificial

Unidade 2: Busca

Unidade 3: Sistemas Baseados em Conhecimento

Unidade 4: Aprendizado de Máquina e Redes Neurais

Unidade 5: Tópicos Especiais



Conteúdo Programático

Unidade 1: Fundamentos de Inteligência Artificial

Unidade 2: Busca

Unidade 3: Sistemas Baseados em Conhecimento

Unidade 4: Aprendizado de Máquina e Redes Neurais

Unidade 5: Tópicos Especiais



Conteúdo Programático

Unidade 1: Fundamentos de Inteligência Artificial

Unidade 2: Busca

2.1. Resolução de Problemas por meio de busca

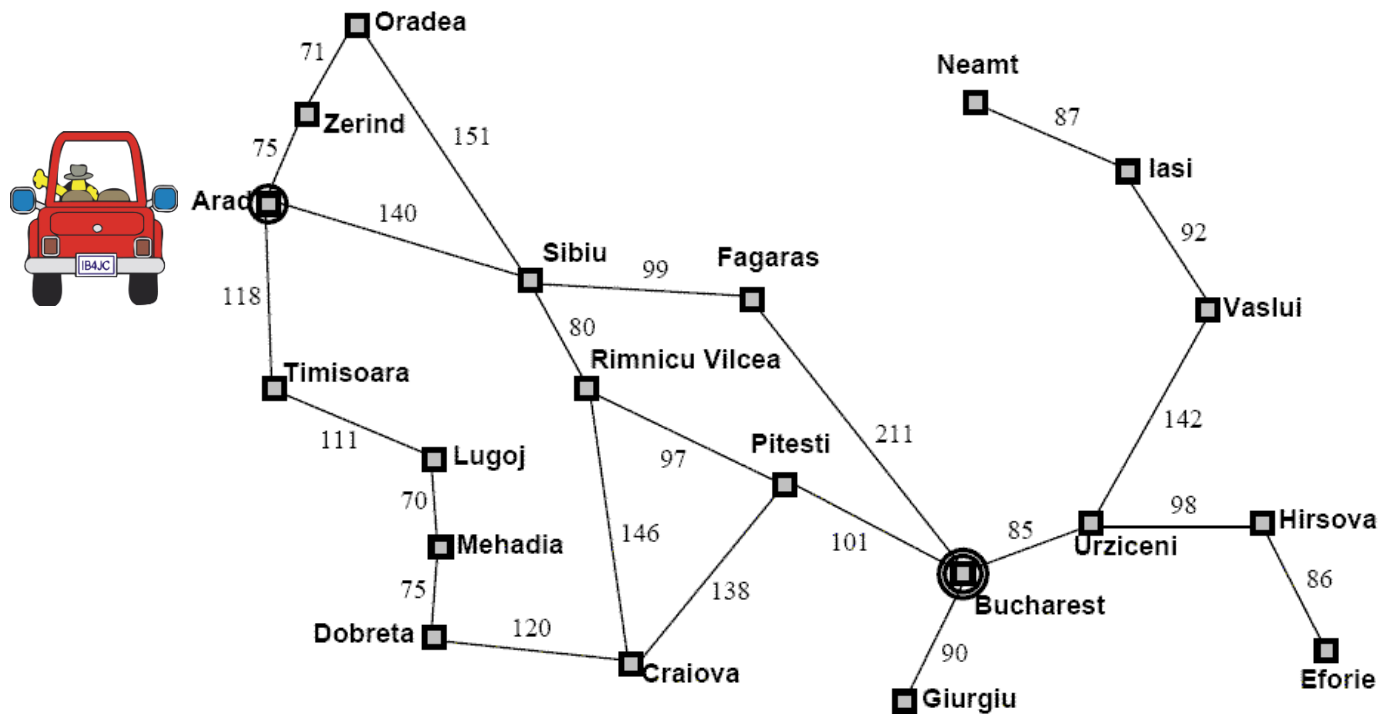
2.2. Busca Cega ou Exaustiva

2.3. Busca Heurística

2.4. Algoritmos Genéticos



Relembrando: Problema de Busca



Relembrando: Definição de um Problema

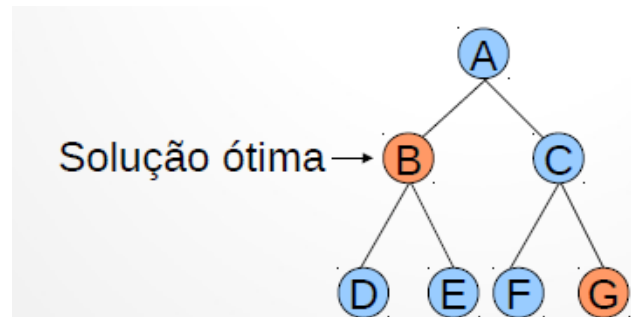
- **Estado Inicial:** Estado inicial do agente.
 - Ex: Em(Arad)
- **Estado Objetivo (Estado Final):** Estado buscado pelo agente.
 - Ex: Em(Bucharest)
- **Ações Possíveis (Função Sucessor):** Conjunto de ações que o agente pode executar.
 - Ex: Ir(Cidade, PróximaCidade)
- **Espaço de Estados:** Conjunto de estados que podem ser atingidos a partir do estado inicial.
 - Ex: Mapa da Romênia.
- **Custo de Caminho:** Custo numérico de cada caminho.
 - Ex: Distância em KM entre as cidades.

Relembrando: Solução para um Problema

- A **solução** para um problema é um caminho desde o estado inicial até o estado objetivo (estado final).
- A qualidade da solução é medida pela função de custo de caminho, isto é, a **solução que tiver menor custo** de caminho entre todas as soluções.

Medida de Desempenho do Algoritmo de Busca

- Uma estratégia de busca é definida pela escolha da **ordem da expansão de nós**
- Estratégias são avaliadas de acordo com os seguintes critérios:
 - **Completeza:** o algoritmo sempre encontra a solução se ela existe?
 - **Otimização (Custo de Caminho):** a estratégia encontra a solução ótima? - Qualidade da solução
 - Para passos com igual custo, é aquela em menor profundidade na árvore de busca



Medida de Desempenho do Algoritmo de Busca

- Uma estratégia de busca é definida pela escolha da **ordem da expansão de nós**
- Estratégias são avaliadas de acordo com os seguintes critérios:
 - **Complexidade De Tempo (Custo de Busca):** quanto tempo ele leva para encontrar a solução? - Número de nós gerados
 - **Complexidade De Espaço (Custo de Busca):** quanta memória é necessária para executar a busca? - Número máximo de nós na memória.

Custo Total

Custo do Caminho + Custo de Busca.

Métodos de Busca

- **Busca Cega ou Exaustiva: (Sem informação)**

- Não tem nenhuma informação adicional sobre os estados, isto é, não sabe qual o melhor nó da fronteira a ser expandido. Apenas distingue o estado objetivo dos não objetivos.

- **Busca Heurística:**

- Ou busca com informação, estima qual o melhor nó da fronteira a ser expandido baseado em funções heurísticas.

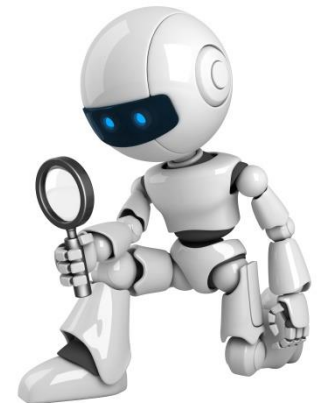
- **Algoritmos Genéticos:**

- Variante de Busca Local em que é mantida uma grande população de estados. Novos estados são gerados por mutação e por crossover, que combina pares de estados da população.

Algoritmos de Busca Cega ou Exaustiva

As estratégias de busca sem informação se distinguem pela **ordem** em que os nós são expandidos.

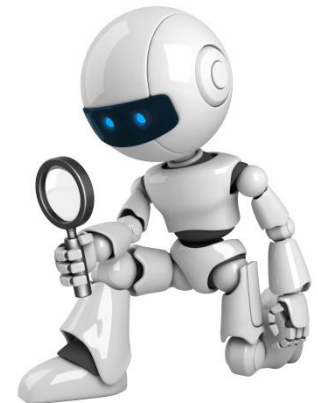
1. Busca em extensão/largura;
2. Busca em profundidade;
3. Busca por aprofundamento iterativo;
4. Busca de custo uniforme.



Algoritmos de Busca Cega ou Exaustiva

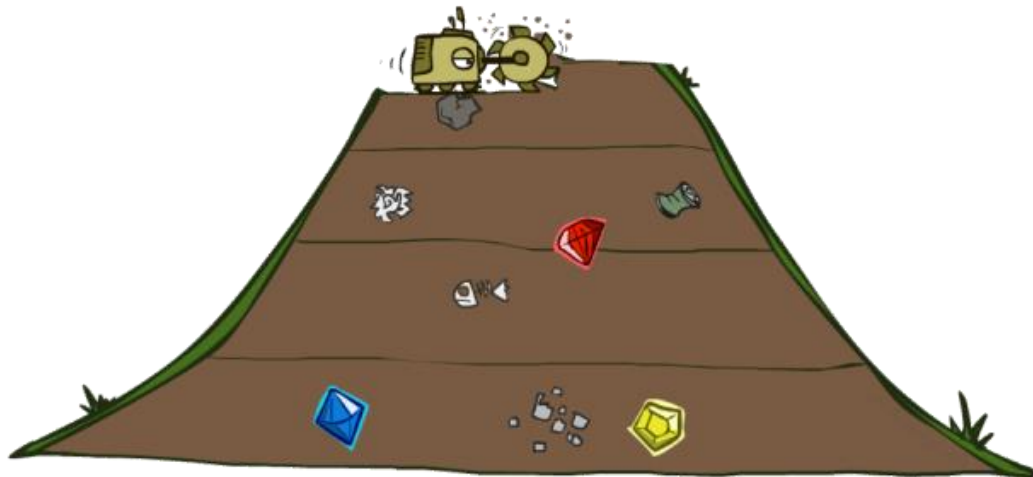
As estratégias de busca sem informação se distinguem pela **ordem** em que os nós são expandidos.

1. Busca em extensão/largura;
2. Busca em profundidade;
3. Busca por aprofundamento iterativo;
4. Busca de custo uniforme.



1. Busca em Extensão / Largura

O nó raiz é expandido primeiro, em seguida todos os nós sucessores são expandidos, então todos próximos nós sucessores são expandidos, e assim por diante.

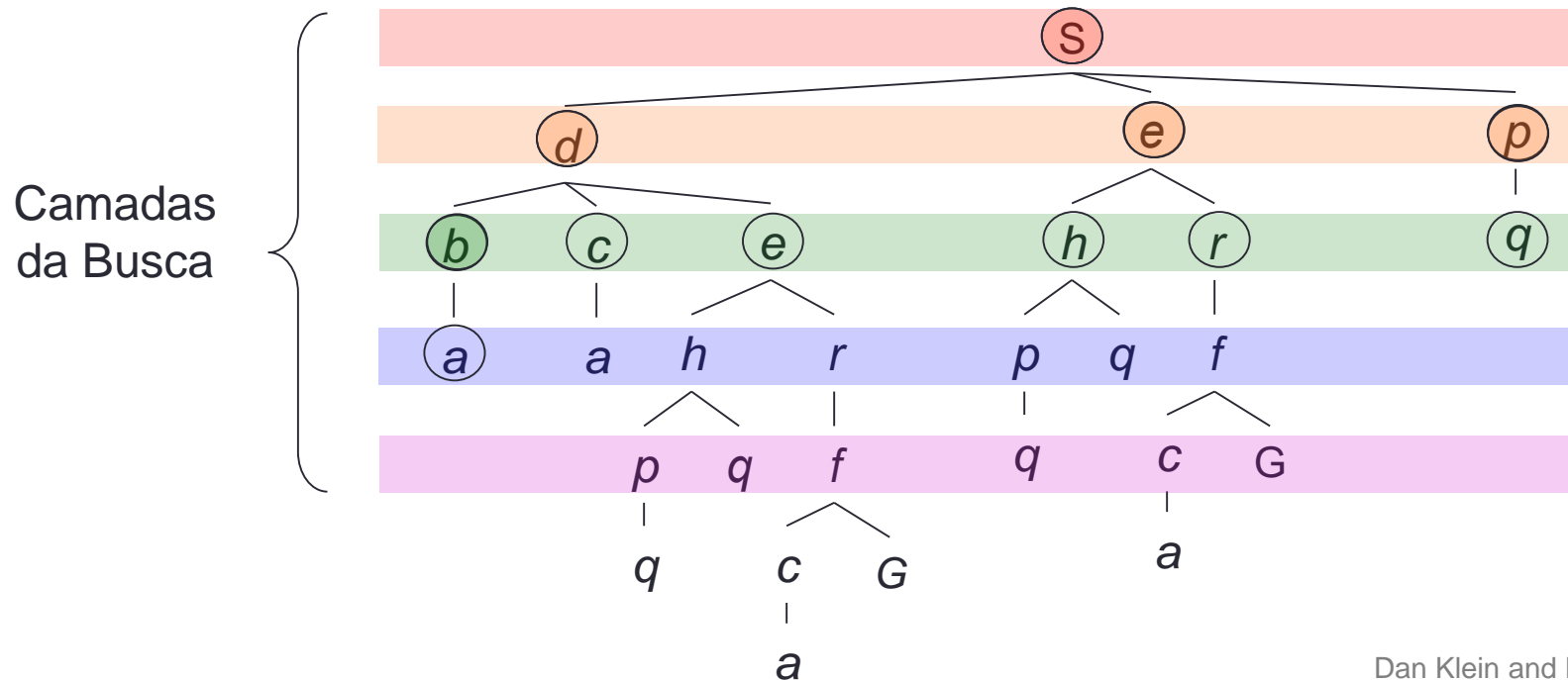


Dan Klein and Pieter Abbeel
ai.berkeley.edu

1. Busca em Extensão / Largura

Estratégia: expandir primeiro o nó mais raso.

Implementação: Utiliza-se uma estrutura de fila (FIFO: first-in-first-out). Os nós visitados primeiro serão expandidos primeiro.



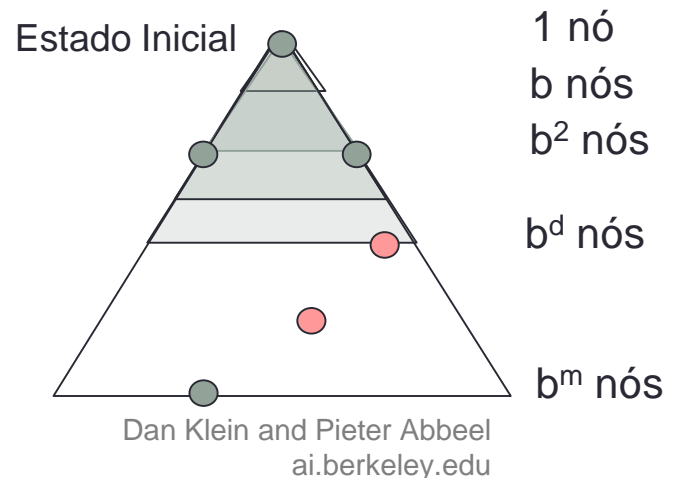
1. Busca em Extensão / Largura

- Completa?
 - Sim, se o fator de ramificação é finito.
- Ótima?
 - Sim, somente no caso em que todas as ações tiverem o mesmo custo.
- Tempo?
 - $1 + b + b^2 + b^3 + \dots + b^d = O(b^d)$
 - A complexidade é exponencial, o que torna impraticável para problemas grandes.
- Espaço?
 - $O(b^d)$ - mantém todos os nós na memória
 - árvore com 12 de profundidade: 1 PB

b: máximo fator de ramificação

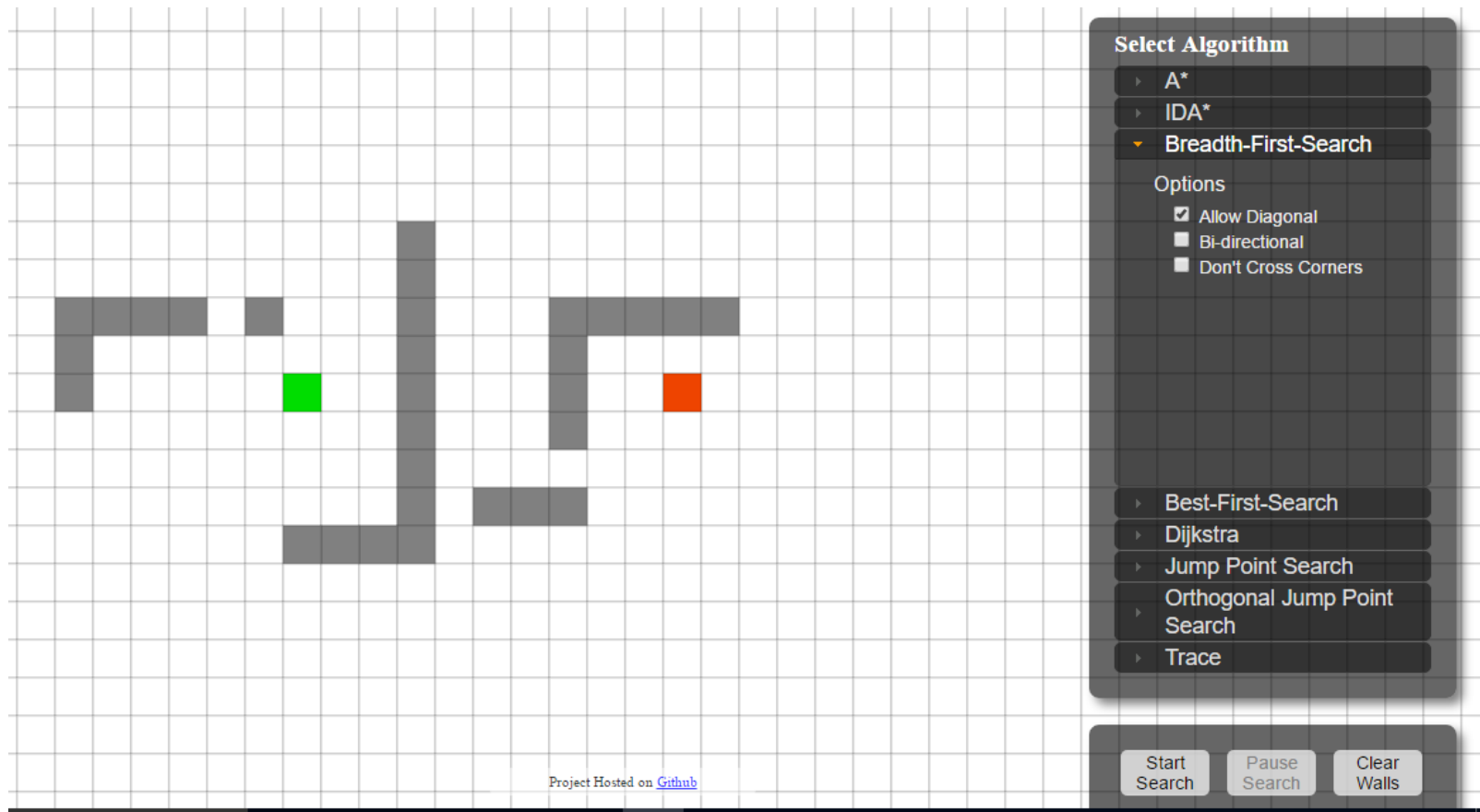
d: profundidade do nó objetivo menos profundo

m: o comprimento máximo (pode ser ∞)



1. Busca em Extensão / Largura

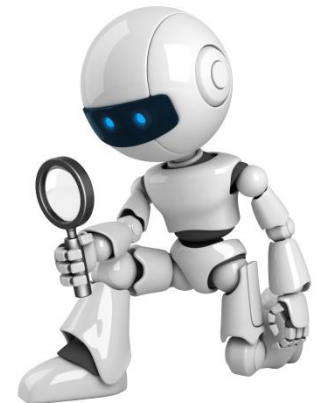
<https://qiao.github.io/PathFinding.js/visual/>



Algoritmos de Busca Cega ou Exaustiva

As estratégias de busca sem informação se distinguem pela **ordem** em que os nós são expandidos.

1. Busca em extensão/largura;
2. Busca em profundidade;
3. Busca por aprofundamento iterativo;
4. Busca de custo uniforme.



2. Busca em Profundidade

A busca prossegue imediatamente até o nível mais profundo da árvore de busca, onde os nós não tem sucessores.

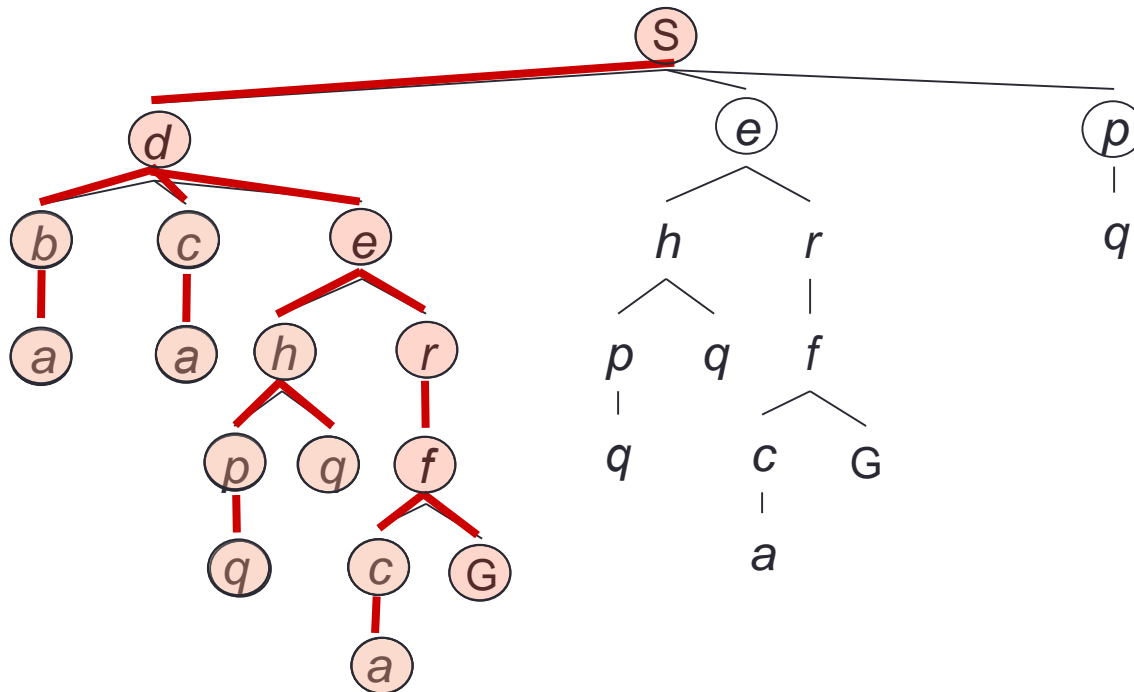


Dan Klein and Pieter Abbeel
ai.berkeley.edu

2. Busca em Profundidade

Estratégia: expande primeiro o nó mais profundo.

Implementação: Utiliza-se uma estrutura de pilha (LIFO - last-in-first-out) para armazenar os nós da fronteira.



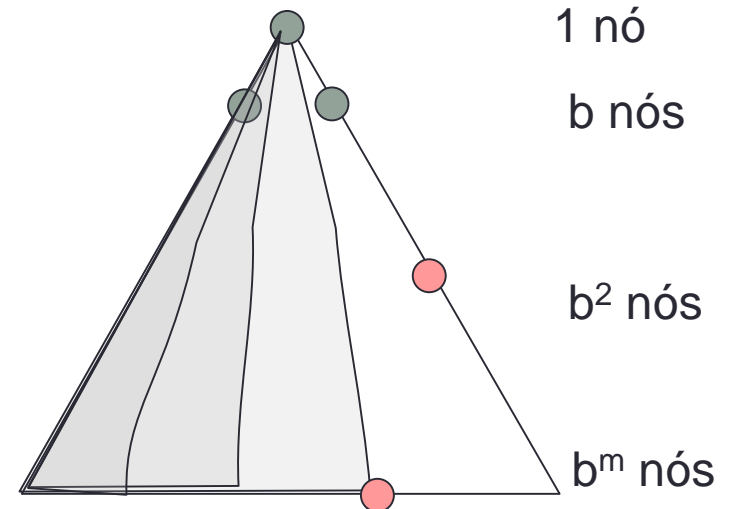
2. Propriedades da Busca em Profundidade

- Completa?
 - Não. Falha em espaços com profundidade infinita, espaços com loops.
 - Se modificada para evitar estados repetidos é completa para espaços finitos.
- Ótima?
 - Não. Encontra a solução "mais à esquerda", independentemente da profundidade ou custo.
- Tempo?
 - $O(b^m)$: Péssimo quando m é muito maior que d .
 - Porém se há muitas soluções pode ser mais eficiente que a busca em extensão.
- Espaço?
 - $O(bm)$
 - Árvore com 12 de profundidade: 118 KB
 - Nós expandidos sem descendentes podem ser removidos.

b : máximo fator de ramificação

d : profundidade do nó objetivo menos profundo

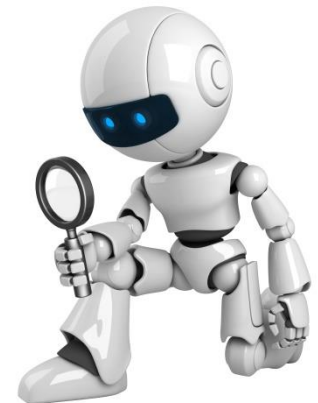
m : o comprimento máximo (pode ser ∞)



Algoritmos de Busca Cega ou Exaustiva

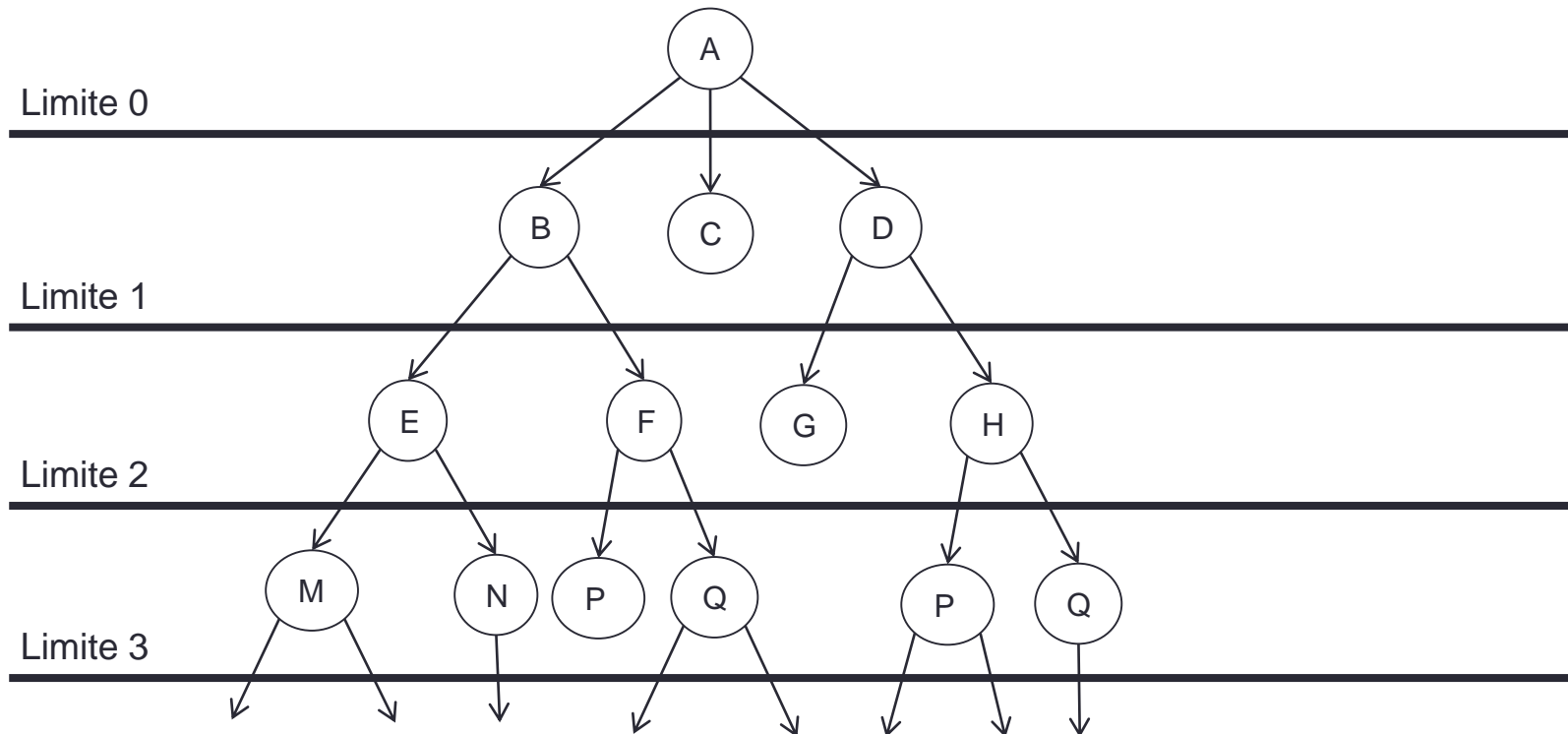
As estratégias de busca sem informação se distinguem pela **ordem** em que os nós são expandidos.

1. Busca em extensão/largura;
2. Busca em profundidade;
3. Busca por aprofundamento iterativo;
4. Busca de custo uniforme.



3. Busca por Aprofundamento Iterativo

- Consiste em uma busca em profundidade onde o limite de profundidade é incrementado gradualmente.



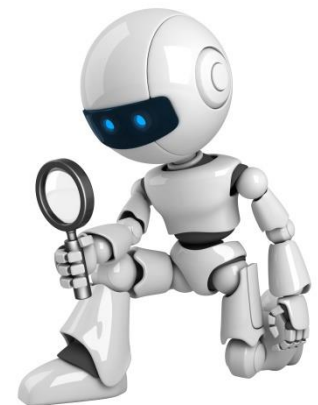
3. Busca por Aprofundamento Iterativo

- Combina os benefícios da busca em profundidade com os benefícios da busca em extensão.
- Evita o problema de caminhos muito longos ou infinitos.
- A repetição da expansão de estados não é tão ruim, pois a maior parte dos nós estão nos níveis inferiores.
- Cria menos estados que a busca em largura e consome menos memória.
- Método de busca sem informação preferido quando existe um espaço de busca grande e a profundidade da solução não é conhecida.

Algoritmos de Busca Cega ou Exaustiva

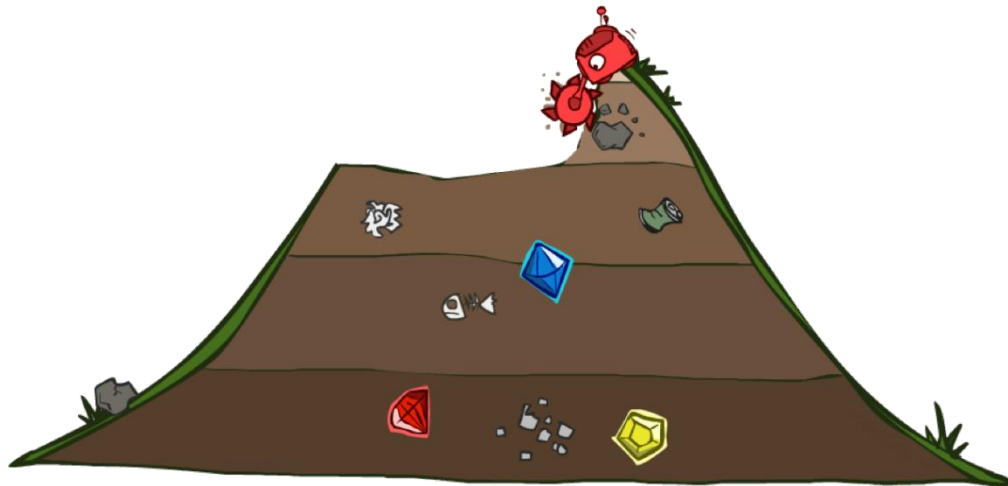
As estratégias de busca sem informação se distinguem pela **ordem** em que os nós são expandidos.

1. Busca em extensão/largura;
2. Busca em profundidade;
3. Busca por aprofundamento iterativo;
4. Busca de custo uniforme.



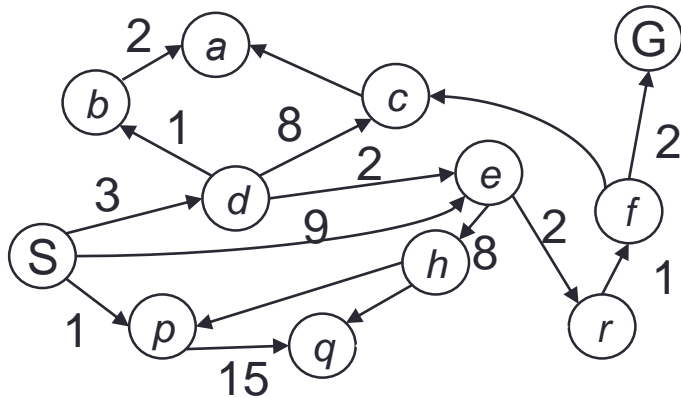
4. Busca de Custo Uniforme

Expande sempre o nó de menor custo de caminho. Se o custo de todos os passos forem os mesmos, a busca será idêntica à busca em extensão/largura.

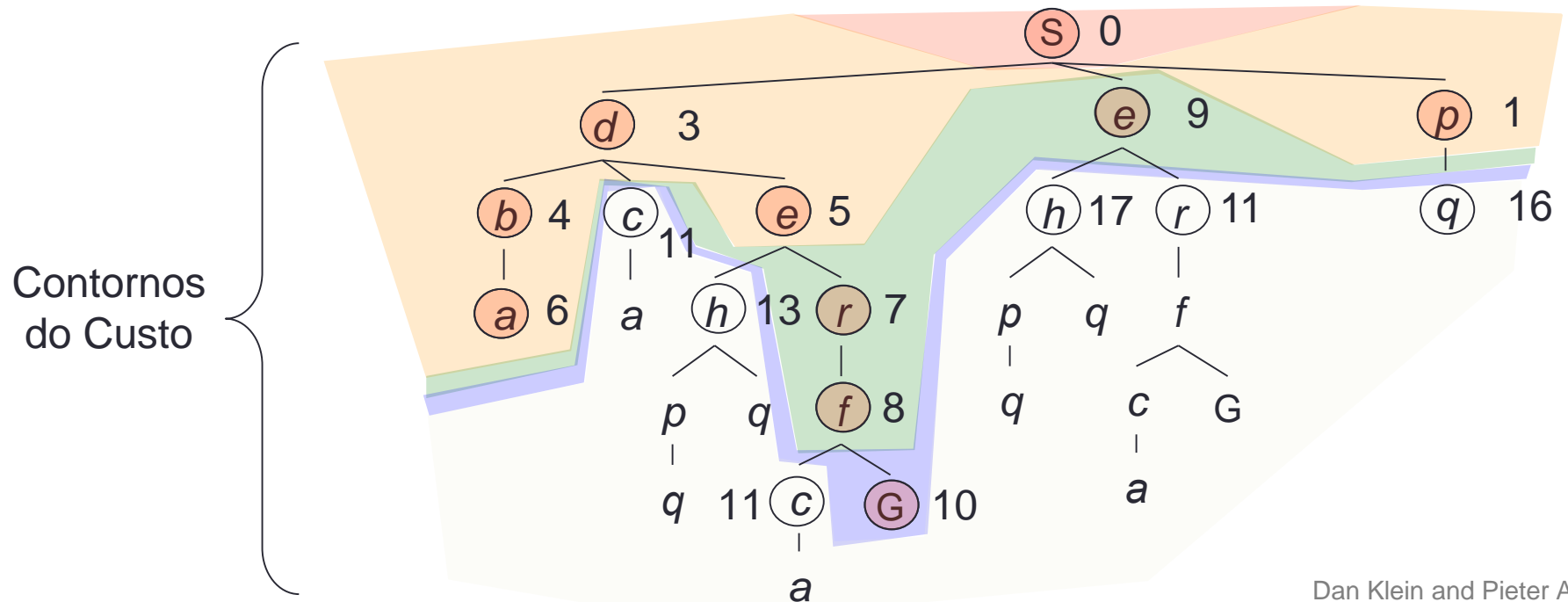


Dan Klein and Pieter Abbeel
ai.berkeley.edu

4. Busca de Custo Uniforme



Estratégia: expandir o nó de menor custo primeiro.



4. Busca de Custo Uniforme

- Não se importa com o número de passos que o caminho tem, apenas com seu custo total.
- A primeira solução encontrada é a **solução ótima** se custo do caminho sempre aumentar ao longo do caminho, ou seja, não existirem operadores com custo negativo.
- Implementação semelhante a busca em largura. Adiciona-se uma **condição de seleção (custo)** dos nós a serem expandidos.

4. Busca de Custo Uniforme

- Completa?
 - Sim, considerando que a melhor solução tem um custo finito e o custo mínimo é positivo.
- Ótima?
 - Sim, pois os nós são expandidos em ordem crescente de custo total.
- Tempo?
 - $O(b^{\lceil C^*/\epsilon \rceil})$ onde C^* é o custo da solução ótima
- Espaço?
 - $O(b^{\lceil C^*/\epsilon \rceil})$
 - Explora pequenos passos antes de explorar caminhos com passos grandes. - exponencial

ϵ : Custo mínimo de uma ação
 C^* : Custo da solução ótima
 b : Nós

