

Inteligência Computacional

Rafael D. Ribeiro, M.Sc.
rafaeldiasribeiro@gmail.com
<http://www.rafaeldiasribeiro.com.br>

Inteligência Computacional

Métodos Revogáveis de Busca

- Busca em profundidade
- Backtracking
- Busca em largura
- Busca em profundidade iterativa
- **Busca ordenada** (busca de custo uniforme)
- **Busca Informada**

Inteligência Computacional

- A busca em largura e a busca em profundidade iterativa sempre encontrarão a melhor solução de um problema se ela existir.
- O conceito de melhor solução utilizado pelos métodos anteriores foi definido como aquela que estava mais próxima da raiz, ou seja, a que foi obtida com a menor quantidade de aplicações de regras.
- A utilização dessas regras não estava associada a qualquer custo. Isto porque nos problemas estudados não havia interesse em se descobrir o custo de uma solução.
- Entretanto, há muitos problemas em que o custo de uma solução desempenha um papel fundamental na sua aceitação.

Inteligência Computacional

- Para esse tipo de problema pode-se aplicar o método chamado de busca ordenada ou busca de custo uniforme.
- Esse método é uma versão mais genérica da busca em largura mas, em vez de expandir os nós menos profundos, os nós expandidos são aqueles que possuem o menor custo de obtenção.
- Para um determinado nó, este custo é o somatório dos custos dos operadores aplicados desde o nó raiz até ele. Dessa maneira, considerando apenas custos não-negativos, os nós filhos sempre terão custos maiores ou iguais que os custos de seus pais.

Inteligência Computacional

- O método inicia expandindo o nó raiz. Se nenhum dos nós filhos for alguma solução, então aquele que tiver o menor custo de obtenção é expandido.
- Se alguma solução ainda não tiver sido encontrada, então de todos os nós abertos, expande-se aquele que tiver o menor custo e assim por diante.
- Ressalta-se que os nós abertos podem estar em qualquer profundidade. A busca ordenada não leva em consideração a posição de um nó, mas sim o seu custo.
- Frequentemente, durante a expansão da árvore, é gerado um nó filho que representa um estado já codificado em um nó anterior. Assim, se o nó anterior já estiver fechado, então não é necessário avaliar o novo nó nem colocá-lo na árvore. Por outro lado, se o nó anterior ainda estiver aberto, deve-se determinar o custo do novo nó e manter aquele que tiver o menor custo.

Inteligência Computacional

- Algoritmo de Busca Ordenada ou de Custo Uniforme

resposta = nulo

lista-de-abertos = estado inicial

lista-de-fechados = nulo

sucesso = falso

enquanto (sucesso = falso) e (lista-de-abertos ≠ vazio) faça

 nó-candidato = elemento de menor custo da lista-de-abertos

 remova o elemento de menor custo da lista-de-abertos

 coloque-o em lista-de-fechados

 se nó-candidato é solução

 então

 sucesso = verdadeiro

 resposta = nó-candidato

 senão

 expanda nó-candidato

 coloque o(s) nó(s) filho(s) na lista-de-abertos

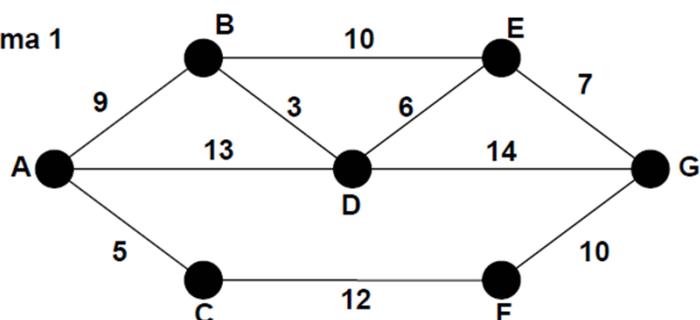
 fim-se

fim-enquanto

retorna sucesso e resposta

Inteligência Computacional

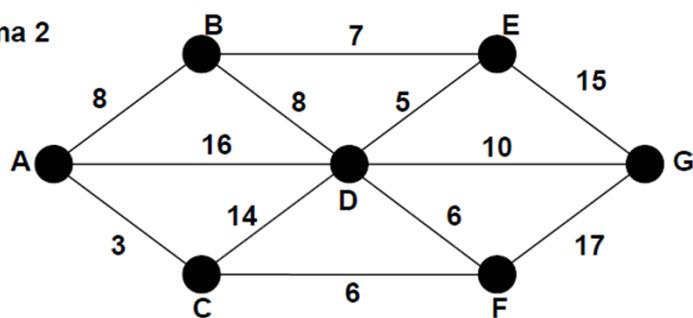
Problema 1



- Caminho entre A e G

Inteligência Computacional

Problema 2



- Caminho entre A e G

Inteligência Computacional

Buscas Informadas

- Os algoritmos anteriores não utilizam qualquer informação do problema que estiver sendo abordado a respeito da proximidade da solução durante a escolha de um nó para a expansão.
- Frequentemente, esses métodos geram e avaliam uma grande quantidade de nós antes de uma solução ser encontrada. Devido aos limites de memória disponível e de tempo para a execução da busca, é desejável que a resolução de um problema seja obtida de uma maneira mais eficiente.

Inteligência Computacional

Buscas Informadas

```
resposta = nulo
lista-de-abertos = estado inicial
lista-de-fechados = nulo
sucesso = falso
enquanto (sucesso = falso) e (lista-de-abertos ≠ vazio) faça
  nó-candidato = elemento de melhor avaliação da lista-de-abertos
  remova o elemento de melhor avaliação da lista-de-abertos
  coloque-o em lista-de-fechados
  se nó-candidato é a solução
    então
      sucesso = verdadeiro
      resposta = nó-candidato
  senão
    expanda nó-candidato
    para cada filho de nó-candidato gerado faça
      avalie-o
      se filho for inédito
        então coloque-o em lista-de-abertos
      senão
        se avaliação nova melhor que antiga
          então
            remova a configuração antiga
            coloque-o em lista-de-abertos
    fim-se
  fim-se
fim-para
fim-se
fim-enquanto
retorna sucesso e resposta
```

Inteligência Computacional

Buscas Informadas

- Os algoritmos que serão vistos a seguir utilizam informações sobre o problema abordado – heurísticas – de modo a selecionar um nó que seja mais promissor para o encontro da solução, evitando que nós presentes em caminhos inúteis sejam selecionados.
- A caracterização de um nó como promissor é realizada pela função de avaliação ou função de avaliação heurística. Quanto menor o valor retornado por esta função para um nó, isto é, quanto menor a sua avaliação, maior a sua possibilidade de estar no melhor caminho do nó raiz até o nó-solução.
- A distinção entre os métodos ocorre na maneira de realizar esta avaliação.

Inteligência Computacional

Buscas Informadas - Busca gulosa

- A estimativa de custo é realizada por uma função chamada de **função heurística** ou apenas **heurística**, sendo representada por $h(n)$.
- Em um dado nó, a função heurística informa o custo estimado do caminho mais barato entre este nó e a solução.
- Um detalhe importante é que o custo estimado do nó-solução deve ser zero, ou seja, $h(\text{solução}) = 0$.
- Uma vez que a busca tenha encontrado o nó-solução, não há mais caminho a seguir e, portanto, não há mais custo. Assim, a avaliação de cada nó é realizada pela função heurística.

Inteligência Computacional

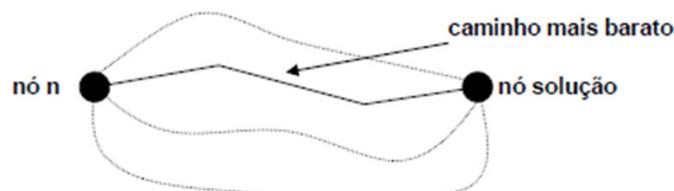
Buscas Informadas - Busca gulosa

- Inicialmente o nó raiz é expandido e seus filhos avaliados segundo uma função heurística. Se nenhum deles for a solução, aquele que tiver o menor custo estimado é expandido e seus filhos são avaliados. Se nenhum desses filhos for a solução, o processo é repetido com um deles, isto é, com o que tiver menor custo estimado e assim por diante. Quando um nó folha é selecionado para a expansão, ele é fechado e a busca continua com o nó aberto de menor avaliação.

Inteligência Computacional

Buscas Informadas - Busca gulosa

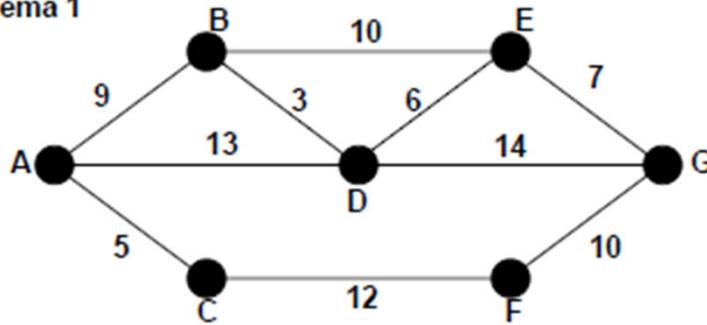
- As estimativas não são necessariamente cálculos exatos dos custos. Por exemplo, uma heurística eficiente em problemas de rotas é a distância em linha reta entre duas localidades.
- É importante observar que quanto mais precisa for a definição de uma heurística melhor será o comportamento da busca. Além disso, a definição de uma função heurística para um problema depende dos pontos de vista de um pesquisador e de suas fontes de informação.



Inteligência Computacional

Buscas Informadas - Busca gulosa

Problema 1

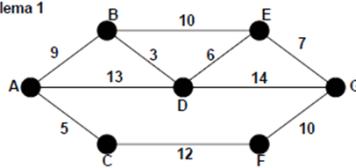


Nós	A	B	C	D	E	F	G
h(nó)	24	15	22	12	7	7	0

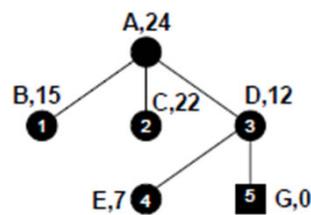
Inteligência Computacional

Buscas Informadas - Busca gulosa

Problema 1



Nós	A	B	C	D	E	F	G
h(nó)	24	15	22	12	7	7	0

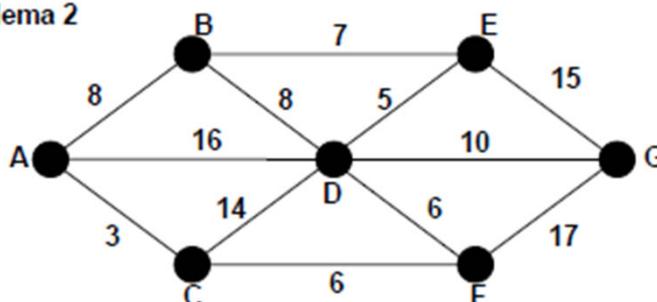


- Considerando o primeiro problema, a busca gulosa encontrou uma solução por meio da rota A-D-G que possui custo igual a 27. Pode-se observar que **esta solução não é ótima**, bastando compará-la a solução obtida pela busca ordenada – rota A-B-D-E-G com custo igual a 25.
- Ainda comparando com a busca ordenada, menos nós foram gerados e avaliados.

Inteligência Computacional

Buscas Informadas - Busca gulosa

Problema 2

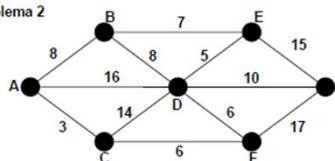


Nós	A	B	C	D	E	F	G
h(nó)	16	13	15	7	10	10	0

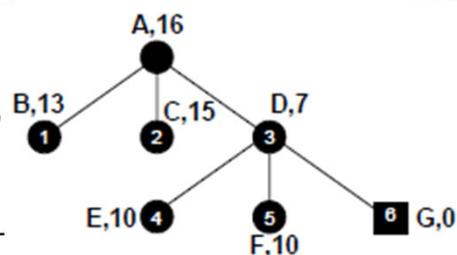
Inteligência Computacional

Buscas Informadas - Busca gulosa

Problema 2



Nós	A	B	C	D	E	F	G
h(nó)	16	13	15	7	10	10	0



- A solução também não foi ótima – rota A-D-G de custo igual a 26 – e, novamente, menos nós foram gerados e avaliados em relação à busca ordenada.

Inteligência Computacional

- Bibliografia utilizada para estas notas de aula:
 - Russel, Stuart J. Inteligência Artificial. Stuart J. Russel. Rio de Janeiro. Elsevier. 2004
 - Notas de Aula do Prof. ROGÉRIO ESPÍNDOLA, disponível na Biblioteca Virtual de docentes – SIA – Estácio, para a disciplina de Inteligência Computacional 1