

Resolução de Problemas por Busca
Busca Informada
Função de Avaliação, Função de Custo, Função
Heurística

Inteligência Artificial – 2020/1

Função de Avaliação na Busca Best-First

- No algoritmo Best-First a função de avaliação pode envolver duas medidas:

- Função de Custo (conhecida a cada passo)

$g(n)$: custo do caminho da raiz até o nó n

- Função Heurística (estimada)

$h(n)$: estimativa de custo do caminho do nó n até o objetivo

Restrição: $h(n) = 0$ quando n é um objetivo

Função de Avaliação na Busca Best-First

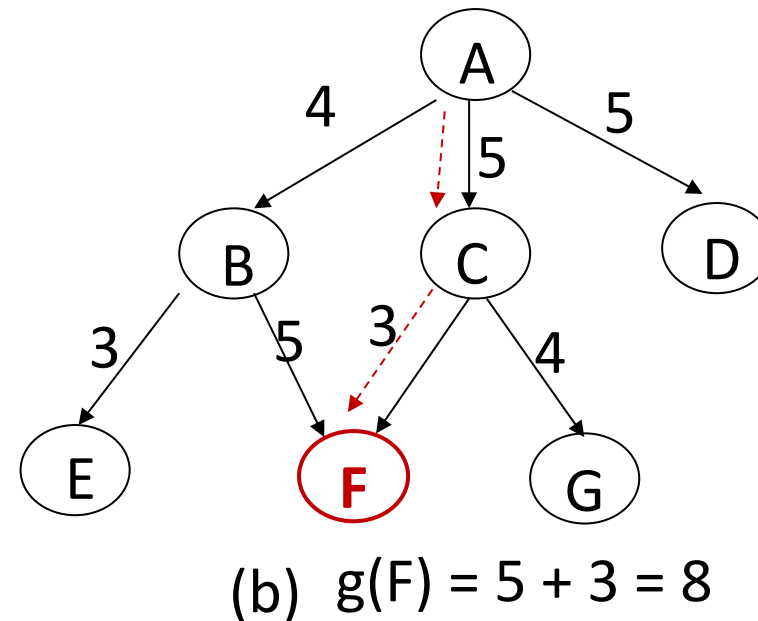
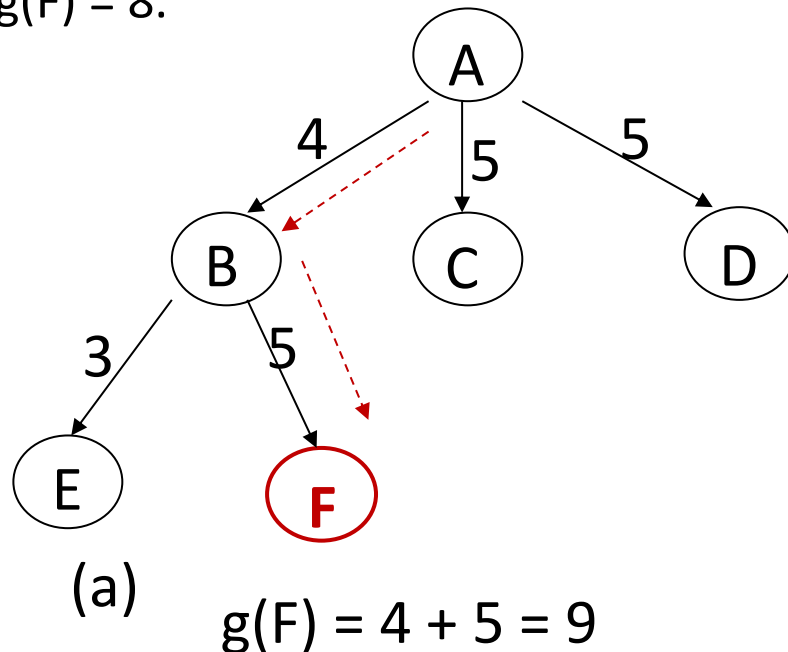
- Função de Custo g
 - Calcula um valor com base nos custos dos movimentos do caminho já percorrido durante o processo de busca, **desde a raiz até o nó**.
 - Um nó n pode ter valores diferentes de g em situações diferentes no mesmo processo de busca, uma vez que o cálculo de g leva em conta o caminho utilizado para chegar até n .

Função de Avaliação na Busca Best-First

Na Figura vemos duas situações, nas quais o nó F teria valores diferentes para a função g:

a) o nó F foi gerado a partir do nó B, levando ao valor de $g(F) = 9$.

b) o nó F pode também ser acessado a partir do nó C, quando o valor de g, para o mesmo nó F, resulta em $g(F) = 8$.



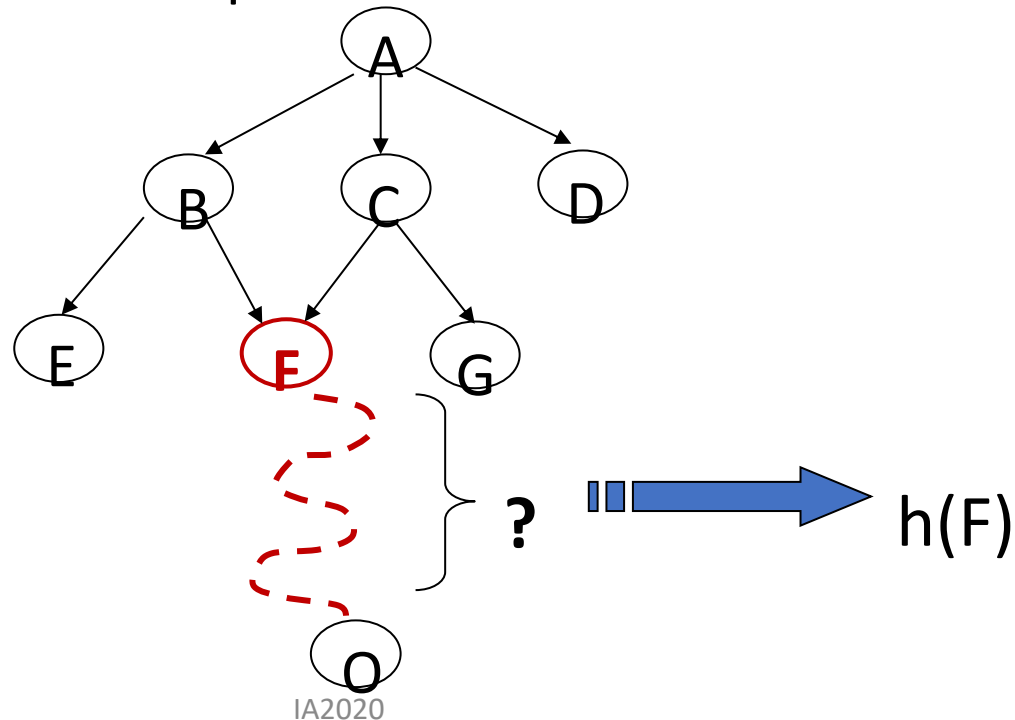
Função de Avaliação na Busca Best-First

- Função de Avaliação Heurística
 - Forma mais comum de aplicar conhecimento adicional do problema ao algoritmo de busca.
 - Definida para cada problema.
 - Denotada por $h(n)$
 - $h(n)$ = custo estimado do caminho mais econômico do nó n até um nó objetivo
 - Restrição: se n é um nó objetivo, então $h(n) = 0$

Função de Avaliação na Busca Best-First

Função de Avaliação Heurística

A função heurística tenta “adivinhar” um valor desconhecido, fazendo a estimativa do custo do caminho que ainda não foi percorrido.



Observações sobre a função heurística

- A função heurística é baseada nas informações do estado em que está sendo aplicada e não considera nenhum tipo de informação relacionada ao custo das operações.
- Na definição de uma função heurística é necessário considerar a eficácia da função e o custo para seu cálculo.
- Uma função heurística mais complexa pode avaliar o estado com mais precisão, mas se o seu cálculo for muito custoso, sua utilização pode ser inviável.

Observações sobre a função heurística

- Não existe procedimento sistemático que possa ser seguido na sua definição.
 - Tentamos aprender como definir uma função heurística para algum problema analisando exemplos conhecidos para outros problemas.
- Como uma heurística é uma forma de “adivinhação” do grau de dificuldade para resolver o problema a partir de um determinado nó, algoritmos que utilizam heurísticas baseiam suas escolhas em chances de obter sucesso, sem garantias de que essas escolhas levarão de fato à melhor solução ou a alguma solução.

Exemplos de funções heurísticas

- Quebra-cabeça de 8 peças
- Diferentes funções heurísticas podem ser definidas para um mesmo problema:
- Número de peças fora do lugar
- Soma das distâncias de cada peça com relação ao seu lugar
- 2x o número de peças em posição reversa
- Combinar 2 e 3

2	8	3
1	6	4
7		5

Estado Inicial

1	2	3
8		4
7	6	5

Estado Final

Exemplos de funções heurísticas

2	8	3
1	6	4
	7	5

2	8	3
1		4
7	6	5

1	2	3
8		4
7	6	5

Estado Final

Peças fora do lugar = 5
Soma das distâncias = 6
2X no. De peças reversas = 0

Peças fora do lugar = 3
Soma das distâncias = 4
2X no. De peças reversas = 0

Exemplos de funções heurísticas

8	2	3
1	6	4
7	5	

1	2	3
8		4
7	6	5

Estado Final

Peças fora do lugar = 4

Soma das distâncias = 4

2X no. De peças reversas = 4

Algoritmos de busca pela melhor escolha

- Variações: dependem da forma da função de avaliação utilizada
- Algoritmo de Busca Gulosa (Greedy)
 - Função heurística
- Algoritmo A^*
 - Função de custo e função heurística

Algoritmos de busca pela melhor escolha

- Algoritmo de Busca Gulosa (Greedy)
 - Minimiza o custo estimado para atingir um objetivo
 - Expande primeiro o nó considerado mais perto do objetivo
 - Função de Avaliação = Função Heurística:

$$f(n) = h(n)$$

onde

- $h(n)$: estimativa heurística do custo do caminho do nó n até o objetivo

Algoritmos de busca pela melhor escolha

- Algoritmo A*
 - Minimiza o custo estimado do caminho que passa por um determinado nó
 - Função de avaliação:

$$f(n) = g(n) + h(n)$$

onde

- n : qualquer estado
- $g(n)$: custo do caminho do nó inicial até o nó n
- $h(n)$: estimativa heurística do custo do caminho do nó n até o objetivo

Busca Gulosa – Exemplo – quebra-cabeça de 8 peças

- Heurística: número de peças fora do lugar

$$f(n) = h(n)$$

2	8	3
1	6	4
7		5

Estado Inicial

1	2	3
8		4
7	6	5

Estado

Final

2	8	3
1	6	4
	7	5

$$f(n) = 4$$

2	8	3
1		4
7	6	5

$$f(n) = 3$$

2	8	3
1	6	4
7	5	

$$f(n) = 5$$

Busca A* – Exemplo – quebra-cabeça de 8 peças

- Heurística: número de peças fora do lugar
- Custo unitário para todos os movimentos

$$f(n) = g(n) + h(n)$$

2	8	3
1	6	4
7		5

Estado Inicial

1	2	3
8		4
7	6	5

Estado
Final

2	8	3
1	6	4
	7	5

$$f(n) = 1 + 4 = 5$$

2	8	3
1		4
7	6	5

$$f(n) = 1 + 3 = 4$$

2	8	3
1	6	4
7	5	

$$f(n) = 1 + 5 = 6$$

- Próxima aula:
- Busca Gulosa e Algoritmo A*