Artificial Intelligence

Assignment 4

Matthias Horbach

mhorbach@uni-koblenz.de

Institute of Web Science and Technologies
Department of Computer Science
University of Koblenz-Landau

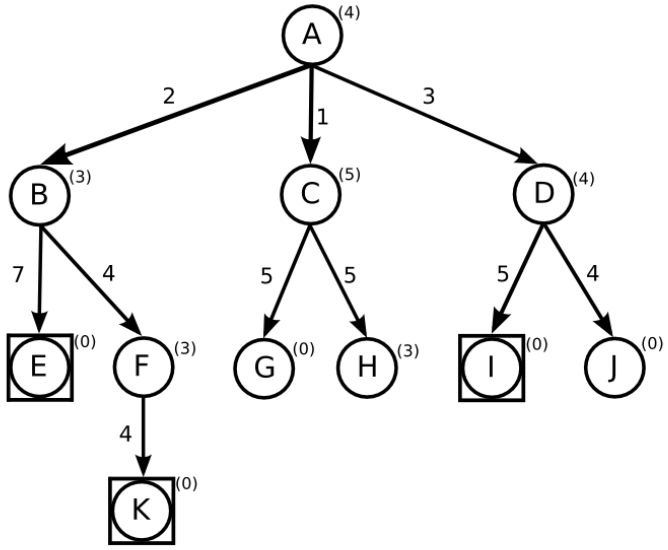
Submission until: 06.06.2020, 23:59

Tutorial on: 07.06.2020

Abhinav Ralhan (221202684), abhinavr8@uni-koblenz.de
Hammad Ahmed (221202832), hammadahmed@uni-koblenz.de
Muhammad Asad Chaudhry (221202659), maac@uni-koblenz.de
Vishal Vidhani (221202681), vvidhani@uni-koblenz.de

1 Uninformed and Informed Search (10 Points)

The following figure shows the complete search tree of a search problem. The start node is node A, all target nodes are marked with a square. The number next to each edge is its cost. A heuristic indicating the estimated distance to the closest target is shown in parentheses next to each node.



For each of the following search strategies, write down (in order) which nodes are visited:

- Breadth first search
- Depth first search
- Iterative deepening with increment 1 (mark where the depth limit increases)
- A*. If two nodes with the same value are present in the frontier, the one which was added to the frontier first will be selected. Provide the f value for all nodes taken from the frontier.
- A*, but the heuristic for D is 8 instead of 4.
  Which solution is optimal? Why does one of the A*-variants not find it?

Solutions:

1. **BFS:**

Index	A	B	C	D	E	F	G	H	I	J	K
Cost	0	2	1	3	9	6	6	6	8	7	10

The path from node A to E:- A, B, C, D , E at weight cost 9

Explored	Frontier
-	A
A	B, C, D
A, B	C, D, E, F
A, B, C	D, E, F, G, H
A, B, C, D	E, F, G, H, I, J
A, B, C, D, E	Goal node Found!!!!

The path from node A to I:- A, B, C, D , E, F, G, H, I at weight cost 8

Explored	Frontier
-	A
A	B, C, D
A, B	C, D, E, F
A, B, C	D, E, F, G, H
A, B, C, D	E, F, G, H, I, J
A, B, C, D, E	F, G, H, I, J
A, B, C, D, E, F	G, H, I, J, K
A, B, C, D, E, F, G	H, I, J, K
A, B, C, D, E, F, G, H	I, J, K
A, B, C, D, E, F, G, H, I	Goal node found!!!!

The path from node A to K:- A, B, C, D , E, F, G, H, I, J, K at weight cost 10

Explored	Frontier
-	A
A	B, C, D
A, B	C, D, E, F
A, B, C	D, E, F, G, H
A, B, C, D	E, F, G, H, I, J
A, B, C, D, E	F, G, H, I, J
A, B, C, D, E, F	G, H, I, J, K
A, B, C, D, E, F, G	H, I, J, K
A, B, C, D, E, F, G, H	I, J, K
A, B, C, D, E, F, G, H, I	J, K
A, B, C, D, E, F, G, H, I, J	K
A, B, C, D, E, F, G, H, I, J, K	Goal Node Found!!!

Complete BFS traverse on this tree will be **A, B, C, D, E, F, G, H, I, J, K.**

2. **DFS:**

The path from node A to E: - only E and we found the goal node because to traverse it will travel the depth of the left child of the root node and the first depth for in order visit is E which is our goal node.
The path from node A to I: - E, B, K, F, G, C, H, A, I
The path from node A to K:- E, B, K, and we found the goal node.
Complete DFS traverse with in-order will be **E, B, K, F, G, C, H, A, I, D, J**

3. **IDS Algorithm:**

The path from node A to E:

Depth	Search Node (In order)
0	A
1	B, C, A, D
2	E <- Goal Node Found!!

The depth limit to find the goal node E is **2.**

The path from node A to I:

Depth	Search Node (In order)
0	A
1	B, C, A, D
2	E, B, K, F, G, C, H, A, I <- Goal Node Found!!!

The depth limit to find the goal node I is **2**.

The path from node A to K:

Depth	Search Node (In order)
0	A
1	B, C, A, D
2	E, B, K <- Goal Node Found!!!

The depth limit to find the goal node I is **2**.

4. **A***

Starting Node is **A**
Explore node A:

Path	G(n)	H(n)	F(n) = g(n) + h(n)
A, B	2	3	5
A, C	1	5	6
A, D	3	4	7

Explore node B:

Path	G(n)	H(n)	F(n) = g(n) + h(n)
A, B, E	9	0	9
A, B, F	6	3	9

Explore node C:

Path	G(n)	H(n)	F(n) = g(n) + h(n)
A, C, G	6	0	6
A, C, H	6	3	9

Explore node G:

No, any successor node for G

Explore node D:

Path	G(n)	H(n)	F(n) = g(n) + h(n)
A, D, I	8	0	8
A, C, J	7	0	7

Explore node J:

No, any successor node for J

Explore node I:

No, any successor node for I

Explore node E:

No, any successor node for E

Explore node F:

Path	G(n)	H(n)	F(n) = g(n) + h(n)
A, B, F, K	10	0	10

Explore node H:

No, any successor node for H

Explore node K:

No, any successor node for K

So, the path for goal nodes will be:

A, B, E => 9
A, D, I => 8
A, B, F, K => 10

5. A* with heuristic for D is 8

Starting Node is **A**

Explore node A:

Path	G(n)	H(n)	F(n) = g(n) + h(n)
A, B	2	3	5
A, C	1	5	6
A, D	3	8	11

Explore node B:

Path	G(n)	H(n)	F(n) = g(n) + h(n)
A, B, E	9	0	9
A, B, F	6	3	9

Explore node C:

Path	G(n)	H(n)	F(n) = g(n) + h(n)
A, C, G	6	0	6
A, C, H	6	3	9

Explore node G:

No, any successor node for G

Explore node E:

No, any successor node for E

Explore node F:

Path	G(n)	H(n)	F(n) = g(n) + h(n)
A, B, F, K	10	0	10

Explore node H:

No, any successor node for H

Explore node K:

No, any successor node for K

Explore node D:

Path	G(n)	H(n)	F(n) = g(n) + h(n)
A, D, I	8	0	8
A, C, J	7	0	7

Explore node J:

No, any successor node for J

Explore node I:

No, any successor node for I

So, the path for goal nodes will be:

A, B, E => 9
A, D, I => 8
A, B, F, K => 10

The result of the BFS and A* is optimal and similar according to the traversing. If we change the heuristic value of any node then the search order is changed.

2 STRIPS (20 Points)

In a rather unorthodox experiment, Prof. Dr. Darksome created raptors which become more intelligent when they drink alcohol. However, as expected by everyone who has ever seen Jurassic Park, two raptors, hans and klaus, managed to free themselves. The professor is currently hiding in a small closet. The door of the closet d_1 is currently closed. However, if a raptor is drunk, he is intelligent enough to open a door. The goal of the two raptors is to open the door. hans is at position a, klaus at position b. Both are standing on the floor, making their height normal. A bottle of whisky, talisker, is on a shelf at at position c—the bottle therefore has a height of high. The door to the closet d_1 is at position d. When raptors drink alcohol, they will drink the entire bottle and then drop it. Therefore, we're ignoring the actual drinking in our formalization. However, a raptor becomes drunk if he drinks alcohol. To reach the shelf, hans and klaus must work together. If one of them bows down, the other one can jump on his back, thereby reaching the bottle on the shelf.

The following predicates are given:

• raptor(X): X is a raptor.
• whisky(X): X is a whisky.
• door(D) D is a door.
• open(X): X is open.
• closed(X): X is closed.
• drunk(X): X is drunk.
• height(X,Y): X has height Y (either normal, low or high).
• pos(X,Y): X is at position Y (either a, b, c or d).
• free(X): Raptor X is free, if no other raptor is standing on his shoulders.
• on(X,Y): Raptor X is standing on raptor Y 's shoulders.

Solutions:

1. Write down the initial STRIPS database and the goal state.

STRIPS database:

- raptor(hans)
- raptor(klaus)
- whisky(talisker)
- door(d1)

- closed(d1)
- height(hans, normal)
- height(klaus, normal)
- height(talisker, high)
- pos(hans, a)
- pos(klaus, b)
- pos(talisker, c)
- pos(d1, d)

Goal state:

- open(d1)

2. Describe the following actions in STRIPS notation.

a) move(R, Xold, Xnew):

C: raptor(R), free(R), pos(R, Xold), height(R, normal)

D: pos(R, Xold)

A: pos(R, Xnew)

b) openDoor(R, D):

C: raptor(R), drunk(R), pos(R, d), closed(D), door(D)

D: closed(D)

A: open(D)

c) drink(R, W):

C: raptor(R), whisky(W), height(R, high), height(R, high), pos(R, a), pos(W, a)

D: whisky(W), pos(W, a), height(W, high)

A: drunk(R)

d) bowDown(R):

C: raptor(R), height(R, normal), free(R)

D: height(R, normal)

A: height(R, low)

e) straightenUp(R):

C: raptor(R), free(R), height(R, low)

D: height(R, low)

A: height(R, normal)

f) climbOn(R1, R2):

C: raptor(R1), raptor(R2), pos(R1, b), pos(R2, b), height(R2, low), height(R1, normal), free(R2)

D: free(R2), height(R1, normal), pos(R1, a)

A: on(R1, R2), height(R1, high)

g) climbDown(R1, R2):

C: raptor(R1), raptor(R2), height(R2, low)

A: height(R1, normal), free(R2)