CS-E4800 Artificial Intelligence
Exercises 3: Search and Planning

J. Rintanen
January 24, 2017

1 Exercise

2 Exercise

3 Exercise

Consider a navigation problem in a grid of size $N \times M$ cells. The goal is to move from cell (1,1) to cell (N,M) by moves north, south, west, and east, and the cost of each action is 1. Clearly, the shortest plans (there are several of them!) have length/cost $N - 1 + M - 1$.

Let us use the the lower bound function $h((x, y)) = \max(N - x, M - y)$ with the A* algorithm. How many states will A* expand, i.e. how many states have an f-value $< N - 1 + M - 1$? How about WA* with $w = 3$?

Solution

We have

$$\begin{aligned} g(x,y) &= (N-1)+(M-1) \\ h(x,y) &= \max(N-x, M-y) \\ f(x,y) &= (N-1)+(M-1)+\max(N-x, M-y) \end{aligned}$$

Only the cells with either $x = N$ or $y = M$ do have an f-value that coincides with the actual cost. For all other cells the f-value is less than the true cost, and they will be expanded.

As an example, consider a 6×6 grid.

g-values:

6	5	6	7	8	9	10
5	4	5	6	7	8	9
4	3	4	5	6	7	8
3	2	3	4	5	6	7
2	1	2	3	4	5	6
1	0	1	2	3	4	5
	1	2	3	4	5	6

h-values:

6	5	4	3	2	1	0
5	5	4	3	2	1	1
4	5	4	3	2	2	2
3	5	4	3	3	3	3
2	5	4	4	4	4	4
1	5	5	5	5	5	5
	1	2	3	4	5	6

$g + h$:

6	10	10	10	10	10	10
5	9	9	9	9	9	10
4	8	8	8	8	9	10
3	7	7	7	8	9	10
2	6	6	7	8	9	10
1	5	6	7	8	9	10
	1	2	3	4	5	6

The cells that have a $g + h$ value < 10 will need to be expanded to prove that the path found is the shortest one. This is all the following.

6						
5	X	X	X	X	X	
4	X	X	X	X	X	
3	X	X	X	X	X	
2	X	X	X	X	X	
1	X	X	X	X	X	
	1	2	3	4	5	6

To reach the goal state (6,6) also one of the value 10 nodes (5,6) or (6,5) need to be expanded. Depending on how *tie-breaking* (arbitrary choice between nodes with the same f-value) turns out, also some other value 10 nodes may be expanded.

Notice that this is a kind of degenerate case and does not demonstrate the advantages of A* well: there is a very high number of shortest solutions, and even a small imperfection in the h-function leads to expanding most nodes. As we will see next, WA* behaves much better in this case.

For WA* with $w = 3$ we have the functions

$$\begin{aligned} g(x,y) &= (N-x)+(M-y) \\ h(x,y) &= 3 \times \max(x-1,y-1) \\ f(x,y) &= (N-x)+(M-y)+2 \times \max(x-1,y-1) \end{aligned}$$

g-values:

6	5	6	7	8	9	10
5	4	5	6	7	8	9
4	3	4	5	6	7	8
3	2	3	4	5	6	7
2	1	2	3	4	5	6
1	0	1	2	3	4	5
	1	2	3	4	5	6

h-values:

6	5	4	3	2	1	0
5	5	4	3	2	1	1
4	5	4	3	2	2	2
3	5	4	3	3	3	3
2	5	4	4	4	4	4
1	5	5	5	5	5	5
	1	2	3	4	5	6

$g + 3h$:

6	20	18	16	14	12	10
5	19	17	15	13	11	12
4	18	16	14	12	13	14
3	17	15	13	14	15	16
2	16	14	15	16	17	18
1	15	16	17	18	19	20
	1	2	3	4	5	6

The WA* search generates nodes with roughly decreasing f-values on and next to the diagonal 15, 16, 14, 15, 13, 14, 12, 13, 11, 12, 10. Essentially, WA* hill-climbs towards the solution, without expanding any nodes outside the solution path.

No node has a value that is less than the cost of the best solution, so once a solution is found, the computation immediately terminates. The solution is an optimal one, but WA* only guarantees that it is at most factor 3 higher than the optimal.

4 Exercise

Let the state variables be $X = \{x_0, x_1, x_2, x_3, x_4\}$. Now the state can be viewed as 4-bit integers $x_4x_3x_2x_1x_0$.

Define actions for incrementing the 5-bit integer (i := i + 1). You need to handle 5 separate cases, when the least significant bits are respectively 0, 01, 011, 0111 and 01111.

Let the initial state be 00000 and the goal 11111. What plan reaches the goal from the initial state?

Note that this is a simple example of a planning problem in which plans can have a length that is exponential in the size of the specification.

Solution

The actions are

$$\begin{aligned} a_0 &= (\{\neg x_0\}, \{x_0\}) \\ a_1 &= (\{\neg x_1, x_0\}, \{x_1, \neg x_0\}) \\ a_2 &= (\{\neg x_2, x_1, x_0\}, \{x_2, \neg x_1, \neg x_0\}) \\ a_3 &= (\{\neg x_3, x_2, x_1, x_0\}, \{x_3, \neg x_2, \neg x_1, \neg x_0\}) \\ a_4 &= (\{\neg x_4, x_3, x_2, x_1, x_0\}, \{x_4, \neg x_3, \neg x_2, \neg x_1, \neg x_0\}) \end{aligned}$$

The plan to reach 01111 from 00000 starts by

$$a_0 \quad a_1 \quad a_0 \quad a_2 \quad a_0 \quad a_1 \quad a_0 \quad a_3 \quad a_0 \quad a_1 \quad a_0 \quad a_2 \quad a_0 \quad a_1 \quad a_0 \quad \cdots$$

5 Exercise