Artificial Intelligence Assignment 1

Anton Buguev BS19-02

16 March 2021

Introduction

In this task we need to find the shortest path from initial point to home on a map $N \times N$ with C number of covid cells with infected zones, D number of doctors and M number of masks which can create protection from covid cells and infected zones.
To find routes let us use 2 algorithms: backtracking and A*.

Algorithm 1: Backtracking

Backtracking uses simple formula: it is checking every cell and every possible path to home. But let us make several improvements to this algorithm to accelerate execution and decrease execution time.

First, if we use map 9×9 with 2 covid cells, 1 doctor and 1 mask, the length of the shortest path has to be less than 18, i.e. less then $2 \cdot 9 = 2 \cdot N$. So if path to the cell that we reached exceeds that value, then it is bad path and absolutely not optimal, so actor will not got there and will choose another cell to go.

Second, when actor reached home first time length of this path is saved, so if other path exceeds that length, then it is also not optimal way and actor will choose another cell to go.

Since first founded path may be not shortest, we need to find all possible routes and find the shortest between them.

Algorithm 2: A*

A* algorithm does not consider all possible routes but tries to go directly to the home. This algorithm uses 2 lists in execution: Open list and Closed list.

- Open list contains all cells that actor can visit and their parent, i.e. from which parent it can come there
- Closed list contains all visited cells and their parents, initially it is empty.

.
Also for each cell algorithm calculates special numbers:

$$F = G + H$$

- G is Chebyshev's distance from initial cell to the current cell
- H is Chebyshev's distance from home to the current cell
- F is sum of G and H

It starts from initial point, adds this cell into Open List, parent for initial cell is that cell itself and G, H and F values are 0.

Then until Open list algorithms makes these actions until actor reaches home:

1. Picks cell with lowest F
2. Adds this cell and its parent into Closed list
3. For each adjacent cell that is not in Closed list:
   - Calculates G, H and F
   - If this cell is not in Open list just add this cell, its parent and values into Open list
   - Otherwise:
     - if G value of existed cell is bigger than G of current cell, algorithms replaces old cell with new and puts new G, H, F values
     - Otherwise does nothing

.
When actor reaches home, current path is saved and algorithm stops.

PEAS description

First, **agent type** is actor who performs actions to safely reach the home. Second, **performance measure** is that actor reaches home safely. Third, **environment** is $N \times N$ map with covid cells and infected cells around them, with doctor ans mask who can make protect actor from covid zones. Actor can move in this environment in all directions but only one cell forward. Fourth, **actuators** of actor are legs that are used to go home. Fifth, actor has **sensor** that is covid predictor which feels that in next cell is infected by covid.

Environment also has properties. First, environment is **partially observable** because actor feel covid only one cell before and does not know where doctor or mask. Second, environment is **multiple agent** because except for actor there are also covid cells with infected zones where actor cannot go without protection, also there are cells with doctor or mask where actor can get protection from covid. Third, environment is **deterministic** because all agents except for actor do not move and do not change their properties, so state of the environment does not change, i.e. actor knows that nothing will happen after his\her steps. Fourth, the environment is **sequential** because next step of actor depends on the step that was made before and also if some of the adjacent cell is infected actor can go there only if doctor or mask were visited, i.e. it depends on previous steps. Fifth, the environment is **static** because nothing happens in the environment while actor is choosing next step. Sixths, the environment is **discrete** because actor moves only one cell forward and after each steps carefully chooses next step. Finally, the environment is **known** because actor knows size of the map, knows that he\she cannot go to the infected cells but can get protection from cell with doctor or mask and that goal is to reach home.

Comparison of algorithms

Let us take Student T-test for comparison of these algorithms. So we need to run algorithm several times and track time for each algorithm and each variant, then find mean value and standard deviation for each variant and then compare each pair of algorithms.
Let us make null hypothesis H_0 that all variants for searching th shortest path are the same.
For experiment, position of 2 covid cells, 1 doctor and 1 mask on map 9×9 will be generated randomly.

Table with results:

Number of the map	Time Backtracking Var 1, ms	Time Backtracking Var 2, ms	Time A* Var 1, ms	Time A* Var 2, ms
1	7	10	1	4
2	18	13	1	3
3	4	5	2	3
4	9	10	4	6
5	15	17	11	13
6	8	9	9	10
7	8	10	1	2
8	6	8	6	7
9	4	6	2	3
10	8	9	2	2
11	17	19	9	11
12	9	11	7	8
13	9	9	2	3
14	8	10	1	3
15	7	9	9	10
16	6	8	1	3
17	6	8	5	6
18	13	16	20	22
19	8	9	2	4
20	14	16	12	13
21	6	7	4	5
Mean	9.04	10.42	5.28	6.71
Standard deviation	3.91	3.61	4.82	4.88
Variance	15.28	13.1	23.25	23.82

.

As we can see, time of execution of 2^{nd} variant is bigger than for 1^{st} variant. It happens because in 2^{nd} variant algorithm does extra computations but it does not affect on the search of shortest path. Since actor does not know position of doctor and mask, he\she still has to go to the cell near covid to check whether there is doctor and mask, so the final result is the same but time differs.

For Student T-test we need to calculate t-value for each pair of variants:

$$t - value = \frac{|M1 - M2|}{\sqrt{\frac{V1}{N} + \frac{V2}{N}}}$$

- $M1$ and $M2$ are mean values
- $V1$ and $V2$ are variances
- N number of samples

Since each algorithm has 21 samples, so degree of freedom is $df = 21 + 21 - 2 = 40$.
Also let us make probability of not rejecting H_0 equals $p = 0.05$.
According to T-table, for $p = 0.05$ and $df = 40$ critical value equals 2.02.

1 Comparison: Backtracking V1 and A* V1

Calculate t-value:

$$t - value = \frac{|9.04 - 5.28|}{\sqrt{\frac{15.28}{21} + \frac{23.25}{21}}} = 2.77$$

So t-value for Backtracking V1 and A* V1 is 2.77 that is bigger than critical value 2.02, thus we reject H_0 that these algorithms are the same.

2 Comparison: Backtracking V2 and A* V2

Calculate t-value:

$$t - value = \frac{|10.42 - 6.71|}{\sqrt{\frac{13.1}{21} + \frac{23.82}{21}}} = 2.8$$

So t-value for Backtracking V2 and A* V2 is 2.8 that is less than critical value 2.02, thus we reject H_0 that these algorithms are the same.

3 Comparison: Backtracking V1 and Backtracking V2

Calculate t-value:

$$t - value = \frac{|9.04 - 10.42|}{\sqrt{\frac{15.28}{21} + \frac{13.1}{21}}} = 1.18$$

So t-value for Backtracking V1 and Backtracking V2 is 1.18 that is less than critical value 2.02, thus we do not reject H_0 that these algorithms are the same.

4 Comparison: A* V1 and A* V2

Calculate t-value:

$$t - value = \frac{|5.28 - 6.71|}{\sqrt{\frac{23.25}{21} + \frac{23.82}{21}}} = 0.95$$

So t-value for A* V1 and A* V2 is 0.95 that is less than critical value 2.02, thus we do not reject H_0 that these algorithms are the same.

Impossible maps

Map is considered to be impossible if actor cannot safely reach home, i.e. actor is blocked by infected zones and cannot visit cell with doctor or mask, so cannot leave blocked zone.

					D			
x	x	x						
x	C	x				H		
x	x	x						
		x	x	x				
		x	C	x				
A		x	x	x				M

As we can see above, actor cannot leave 2×3 zone and thus cannot reach home.

Or another example is when home, doctor and mask are blocked by infected cells, so actor also cannot reach home.

				x	x	x	D	H
				x	C	x		M
				x	x	x		
						x	x	x
						x	C	x
						x	x	x
A								

Actually, these examples are the same, i.e. actor is separated from home, doctor and mask by infected cells but time of execution differs. In first example program quickly checks all possible ways and writes that there is no way home, however, in second variant checking of all possible ways takes much more time.

Conclusion

The comparison of 2 algorithms for searching the shortest path was done. And by statistical analysis using Student T-test we can conclude that A* is better almost in all cases than backtracking. However, in some cases A* may give one step longer path because it has several adjacent nodes to choose from and it may choose cell that is not the closest to the home.