CS 520 Introduction to Artificial Intelligence
Homework 1

Malike Alikhani (netid: ma1195)
Ana Echavarra Uribe (RUID: 167007648)

October 12, 2015

Part 0

We designed and algorithm to create a maze like structure based on the algorithm presented in (`\http://www.migapro.com/depth-first-search/`) but with certain modifications. The idea is that when running a depth first search from a given cell select randomly one of its unseen neighbors that are two cells away in any direction (north, south, east or west). Move to that cell but blocking all of the unseen neighbors of the intermediate cell (the cell in between the current cell and the selected neighbor). This creates a maze like structure with no loops and one connected component.
Now, in order to add loops and creating more than one connected component we simply set random blocked cells as unblocked and random unblocked cells as blocked.

Part 2

The repeated forward A* was executed on 50 grid of size 101×101 using two different rules for selecting the next cell to expand when their $f = g + h$ value is the same: selecting the cell with the smaller g value or with the larger value of g. The average number of cells expanded, searches (repetitions of the A* search), number of cells the agent moves and running time are shown in results in table 1.

	Num cells expanded	Num searches	Num moves	Running time (sec)
Smaller g	66,695	97	385	0.1608
Larger g	8,164	95	376	0.0939
Ratio Smaller/Larger	8.1699	1.0185	1.0240	1.7124

Table 1: Average statistics when running on 50 grids and breaking ties in favor of larger and smaller values of g.

It can be seen that, on average, the number of cells expanded when selecting the cell with the smaller value of g is more than 8 times higher than the number of cells expanded when selecting the cells with the larger value of g. The running time, nonetheless is not 8 times higher; this could be because the time overhead of starting a search is so big that the ratio in the running time is only 1.7 times higher even if the search is doing 8 times the work.
This difference in the number of cells expanded could be due to the fact than selecting a cell with a smaller g rather than a cell with a larger g implies that we are favoring the cell with the largest h value. Since the h value is a lower bound on the number of steps from that cell to the target, the cell that is being expanded is possibly farther away from the target than the other cell is. This could in the long run make us explore all the neighbors of that cell which are farther away and are not getting us to closer to the target.

Part 3

The algorithms for the forward and backward A* were executed on the same 50 grids of size 101×101 and using the same initial cell for the agent and for the target cell. The results in terms of the average number of expanded cells, number of searches, number of moves of the agent and running time, along with the ratio comparison of these values are shown in table 2.

	Num cells expanded	Num searches	Num moves	Running time (sec)
Backward A*	455,896	96	373	0.7797
Forward A*	8,164	95	376	0.0879
Ratio Backward/Forward	55.845	1.0103	0.9923	8.8687

Table 2: Average statistics when running on 50 grids and running backward and forward A*.

	Num cells expanded	Num searches	Num moves	Running time (sec)
Adaptive A*	6,062	96	378	0.0886
Repeated A*	8,164	95	376	0.0893
Ration Adaptive A*/Repeated A*	0.7426	1.0078	1.0068	0.9923

Table 3: Average statistics when running on 50 grids and running Repeated A* and adaptive A*.