**Assignment 2 Experiment**

**User:hluu**

**StudentID:1025379**

**Name: Hoang Anh Huy Luu**

**Introduction:**

For this project, I have to build an AI agent based on Dijkstra’s algorithms to solve the Pacman game. Dijkstra’s algorithm is also known as greedy algorithm as it based on the idea that any sub-path along a shortest path is also a shortest path. Using this algorithm, I am able to find all the possibility paths from the initial node and thus, can update the value of the maximum scores to return the best path which is lead to the highest score. To decide which path will lead to the highest score, I will use 2 ways: max and avg methods with the help of breadth – first search to sort the priority which node should be explored first. For max method, I will use the accumulated reward which is updated from the new node up to the first move and then propagate back to this first node to update the maximum of score of any of its children. For the avg method, I also use the accumulated reward to propagate back the first node but take the average score of all of its children.

For this experiment I will have a table with the mean score, mean expanded/second and total execution time for each type of propagation (max/avg) which the budget is 10,100,1000,2000 and also the graph. Because of the random movement of the ghost, I will run each test 3 times for each of 3 level.

**Level 1:**

Max method:

|  |  |  |  |
| --- | --- | --- | --- |
| Budget | Mean score | Mean expanded/second | Total execution time |
| 10 | 190.3 | 149.9 | 192.8 |
| 100 | 294 | 1326.3 | 189.93 |
| 1000 | 294.3 | 11795.2 | 185.23 |
| 2000 | 294 | 19209.1 | 245.1 |

Avg method:

|  |  |  |  |
| --- | --- | --- | --- |
| Budget | Mean score | Mean expanded/second | Total execution time |
| 10 | 112.6 | 121.5 | 128.8 |
| 100 | 56.7 | 1224.3 | 123.1 |
| 1000 | 149 | 10427.6 | 208.8 |
| 2000 | 110.3 | 18806.1 | 153.53 |

**Level 2:**

Max method:

|  |  |  |  |
| --- | --- | --- | --- |
| Budget | Mean score | Mean expanded/second | Total execution time |
| 10 | 318.6 | 121.2 | 131.4 |
| 100 | 646 | 1196.3 | 180.6 |
| 1000 | 400 | 10229.6 | 200.3 |
| 2000 | 580.6 | 18946.3 | 225.4 |

Avg method:

|  |  |  |  |
| --- | --- | --- | --- |
| Budget | Mean score | Mean expanded/second | Total execution time |
| 10 | 78.3 | 122.2 | 42.9 |
| 100 | 189.3 | 1212.3 | 107.3 |
| 1000 | 135.3 | 10213.4 | 107.1 |
| 2000 | 104.6 | 19233.4 | 117.3 |

**Level 3:**

Max method:

|  |  |  |  |
| --- | --- | --- | --- |
| Budget | Mean score | Mean expanded/second | Total execution time |
| 10 | 474.3 | 128.3 | 228.2 |
| 100 | 533.6 | 1201.2 | 196.3 |
| 1000 | 429 | 12917.3 | 254.6 |
| 2000 | 452 | 27793.2 | 229.2 |

Avg method:

|  |  |  |  |
| --- | --- | --- | --- |
| Budget | Mean score | Mean expanded/second | Total execution time |
| 10 | 232 | 130.7 | 225.4 |
| 100 | 153 | 1156.6 | 275.8 |
| 1000 | 267 | 11564.7 | 298.2 |
| 2000 | 245 | 25586.7 | 217.5 |

**Discussion:**

Based on the observation of all the table and graph above. I can see that using the max budget method always have the higher score than the average method. Moreover, in the max method, if I increase the size of the budget, it will increase the score also as the AI agent will explored deeper in all the paths. On the other hand, the average method seems not very accurate because it takes the average of all of the children nodes so it also includes the possibility of bad behavior.

In addition, in regards of time complexity of Dijkstra’s algorithm, it will take O((V+E)log(V)) and it is also confirmed according to the tables above. However, sometime, it is not correct because in the Pacman, the ghosts move randomly so it may affect the results.