Project 01

Aravindh Kuppusamy, Deepak Sharma, Karan Jariwala
CSCI 630-01 Foundations of Intelligent Systems

October 7, 2016

1 Problem Definition

1.1 State

Configuration of all possible positions in the maze along with the dice orientation at each possible position. If the width of the maze is W and height of the maze is H, then W*H positions are possible. Out of these positions, let O positions be obstacles and hence, the dice can't be moved to those locations. 20 orientations of dice are possible for each non-goal position and 4 orientations for goal position. Hence the size of the state space is:

$$\textbf{StatespaceSize} : (((\textbf{W} * \textbf{H}) - \textbf{O} - \textbf{1}) * \textbf{20}) + (\textbf{1} * \textbf{4})$$

1.2 Successor Function

Successor function generates all the possible moves from a given state. Here, Successor function generates states which represents both the dice location in the maze and its orientation. So, the successor function for the problem is legal moves in both maze and dice in same direction.

1.2.1 Legal positions in Maze for successor states

The possible moves include moving in four directions (Top, Bottom, Right, and Left) to the adjacent position with step cost as 1. But if the resulting position is an obstacle, then the move is not legal. So the legal moves would be all possible moves without illegal ones.

1.2.2 Legal orientations of Dice for successor states

The possible orientations include turning in four directions (Top, Bottom, Right, and Left) by one step. But if the resulting position is an orientation of dice with top value = 6 for a non-goal state and top value != 1 for goal state, then the move is not legal. So the legal moves would be all possible moves without illegal ones.

1.3 Goal test

Whether the Goal position in the maze has been reached with the dice orientation having top value = 1.

1.4 Path Cost

Total number of moves till the current state.

2 Heuristics

2.1 MANHATTAN DISTANCE

Manhattan distance is the sum of the vertical and horizontal distances between two points. If (x1, y1) and (x2, y2) are the two points then Manhattan distance between them is

$$\mathbf{d = |x_1 - x_2| + |y_1 - y_2|} \tag{1}$$

Since the problem state is a grid and we cannot move in other than the 4 directions, and by also relaxing the constraint of obstacles through which we cant pass, Manhattan distance is the shortest path and hence its optimal.

2.1.1 Proof for Admissibility

A heuristic is admissible if estimated cost is never greater than the actual cost from the current state to goal state[1].

For a given state at location $(\mathbf{x_s, y_s})$ and goal location $(\mathbf{x_g, y_g})$, the distance we have to travel is at least $|\mathbf{x_s - x_g}| + |\mathbf{y_s - y_g}|$ since we can move only in horizontal and vertical directions. We are also relaxing the constraint of obstacles and dice configurations. Proof by Contradiction:

Base Case:

$$\mathbf{h(goal) = 0}$$

Let C be the minimum actual cost for traveling from start state to goal state. .i.e., at least C steps are required for traveling from start state to goal state Assume, the actual cost(C):

$$\mathbf{C < h(start)} \tag{2}$$

For each action we perform, the step cost will be just 1. So heuristics for the successor state will reduce by just 1.

$$\mathbf{Cost(parent-> child) = 1}$$

Hence,

$$\mathbf{h(successor) = h(parent) - 1}$$

Since, at least C steps are required for traveling from start state to goal state,

$$\mathbf{h(goal) >= h(start) - C}$$

$$h(goal) >= h(start) - C > 0$$

From 2

$$h(goal) > 0$$

This contradicts the base case and so our assumption is wrong. Therefore,

$$C >= h(start)$$

Hence, we conclude that Manhattan distance heuristics is admissible.

2.2 DIAGONAL DISTANCE

Diagonal distance is the sum of the number of diagonal steps possible and the remaining straight line distance between two points. In diagonal Heuristic, the cost is generated while considering the horizontal, vertical and diagonal paths toward the goal which means that it can look up the path or a state in 8 directions while calculating the heuristic cost. If (x_1, y_1) and (x_2, y_2) are the two points then Diagonal distance between them is:

$$d_x = |x_1 - x_2|$$

$$d_y = |y_1 - y_2|$$

$$d = \sqrt{2} * \min(d_x, d_y) + |d_x - d_y| \tag{3}$$
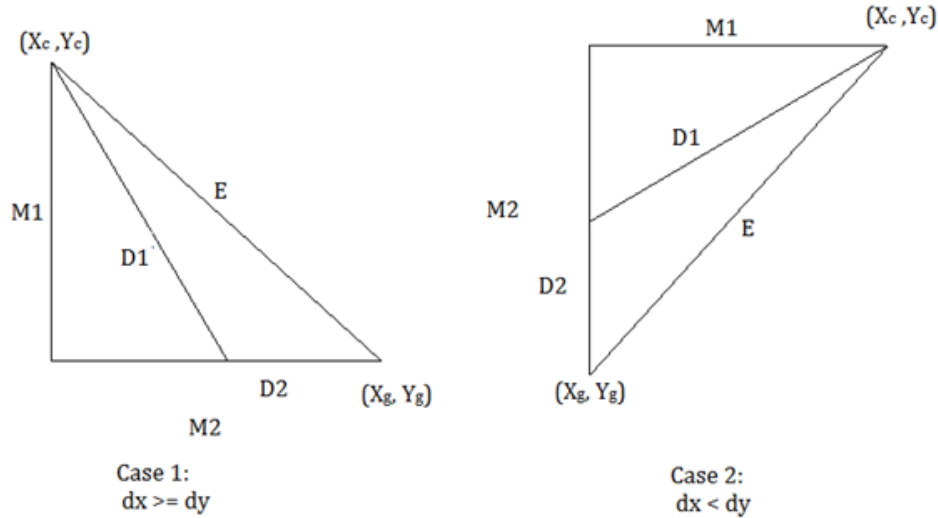
2.2.1 Proof for Admissibility

Figure 1: Left triangle represents the case when $dx >= dy$ and right triangle represents the case when $dx < dy$

3

For a given set of positions (x_1, y_1) and (x_2, y_2), there are two cases possible as shown in the figure 1:

1) $dx >= dy$ or

2) $dx < dy$

For the given current node (x_c, y_c) and the goal node (x_g, y_g), the diagonal heuristic $h_{diagonal} = D1 + D2$. Now by triangle inequality,

$$M1 + (M2 - D2) >= D1$$

$$M1 + M2 >= D1 + D2.$$

Here, $M1 + M2$ is the Manhattan heuristics for the point (x_c, y_c). So, Euclidean distance is always lesser than or equal to Manhattan distance. From above statement, we can say that for any point P in real coordinate system,

$$h_{diagonal}(P) <= h_{manhattan}(P)$$

i.e., Manhattan heuristics dominates diagonal heuristics. Since, Manhattan heuristics dominates diagonal heuristics and also Manhattan heuristics is admissible, we can infer that diagonal heuristics is also admissible.

2.3 EUCLIDEAN DISTANCE

Euclidean distance is the distance of the straight line that connects two points. If (x_1, y_1) and (x_2, y_2) are the two points then Euclidean distance between them is:

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \tag{4}$$

2.3.1 Proof for Admissibility

According to triangle inequality, the sum of two sides is always greater than or equal to the third side.

Euclidean distance is the hypotenuse (third side) in the triangle formed by two sides dx and dy in Manhattan distance, where

$$dx = |x_1 - x_2| \quad \text{and} \quad dy = |y_1 - y_2|$$

So, Euclidean distance is always lesser than or equal to Manhattan distance. From above statement, we can say that,

$$h_{euclidian}(P) <= h_{manhattan}(P)$$

i.e., Manhattan heuristics dominates Euclidean heuristics. Since, Manhattan heuristics dominates Euclidean heuristics and also Manhattan heuristics is admissible, we can infer that Euclidean heuristics is also admissible.

2.4 FANCY MANHATTAN DISTANCE

Fancy Manhattan heuristics is nothing but sum of Manhattan distance and a reward $\alpha = -0.5$ for a state. If the current state's position is $(\mathbf{x_c}, \mathbf{y_c})$ and the goal state's position is $(\mathbf{x_g}, \mathbf{y_g})$, then horizontal and vertical distances between them are

$$\mathbf{dx} = |\mathbf{x_g} - \mathbf{x_c}| \quad \text{and} \quad \mathbf{dy} = |\mathbf{y_g} - \mathbf{y_c}|$$

Reward comes into play if $\mathbf{dx} <= \mathbf{1}$ or $\mathbf{dy} <= \mathbf{1}$. We add reward to heuristics if we rotate the dice from current location $(\mathbf{x_c}, \mathbf{y_c})$ to the goal location $(\mathbf{x_g}, \mathbf{y_g})$ and if it is passing goal test. We perform four types of rotations for calculating dice face value:
　If $\mathbf{dy} <= \mathbf{1}$ then:

1. if $\mathbf{dx} > \mathbf{0}$ then, rotate the dice $|\mathbf{dx}|\mathbf{mod4}$ time in right direction

2. if $\mathbf{dx} < \mathbf{0}$ then, rotate the dice $|\mathbf{dx}|\mathbf{mod4}$ time in left direction.

Now, Rotate the dice by dy steps in corresponding direction.
　Similarly if $\mathbf{dx} <= \mathbf{1}$ then:

1. if $\mathbf{dy} > \mathbf{0}$ then rotate dice $|\mathbf{dy}|\mathbf{mod4}$ times in south direction

2. if $\mathbf{dy} < \mathbf{0}$ or $|\mathbf{dy}|\mathbf{mod4}$ times in North direction.

Now, Rotate the dice by dx steps in corresponding direction.
　Note: This operation is not computational heavy since maximum value by mod 4 operation we can get is 3.
　After performing rotations, If face value is 1 then :

$$\mathbf{h_{fancyManhattan}(n) = h_{manhattan}(n) + \alpha}$$

For other cases:

$$\mathbf{h_{fancyManhattan}(n) = h_{manhattan}(n)}$$

2.4.1 Proof for Admissibility

Here, we are actually either subtracting the Manhattan heuristics by 0.5 as per the above conditions or we just return Manhattan heuristics. So, the fancy Manhattan heuristics is less than or equal to Manhattan heuristics. We know that Manhattan distance is admissible. Hence, the fancy Manhattan heuristics is also admissible since it never over predicts the cost than Manhattan heuristics.

3 Performance metrics

	map1				map2				map3				map4				map5			
	Fancy Manhattan	Manhattan	Euclidean	Diagonal	Fancy Manhattan	Manhattan	Euclidean	Diagonal	Fancy Manhattan	Manhattan	Euclidean	Diagonal	Fancy Manhattan	Manhattan	Euclidean	Diagonal	Fancy Manhattan	Manhattan	Euclidean	Diagonal
Node Generated	12	20	16	16	49	51	65	63	3	3	3	3	77	96	107	117	388	509	1166	1164
Node Visited	8	14	11	11	40	42	51	49	3	3	3	3	63	78	86	96	256	326	1036	1014

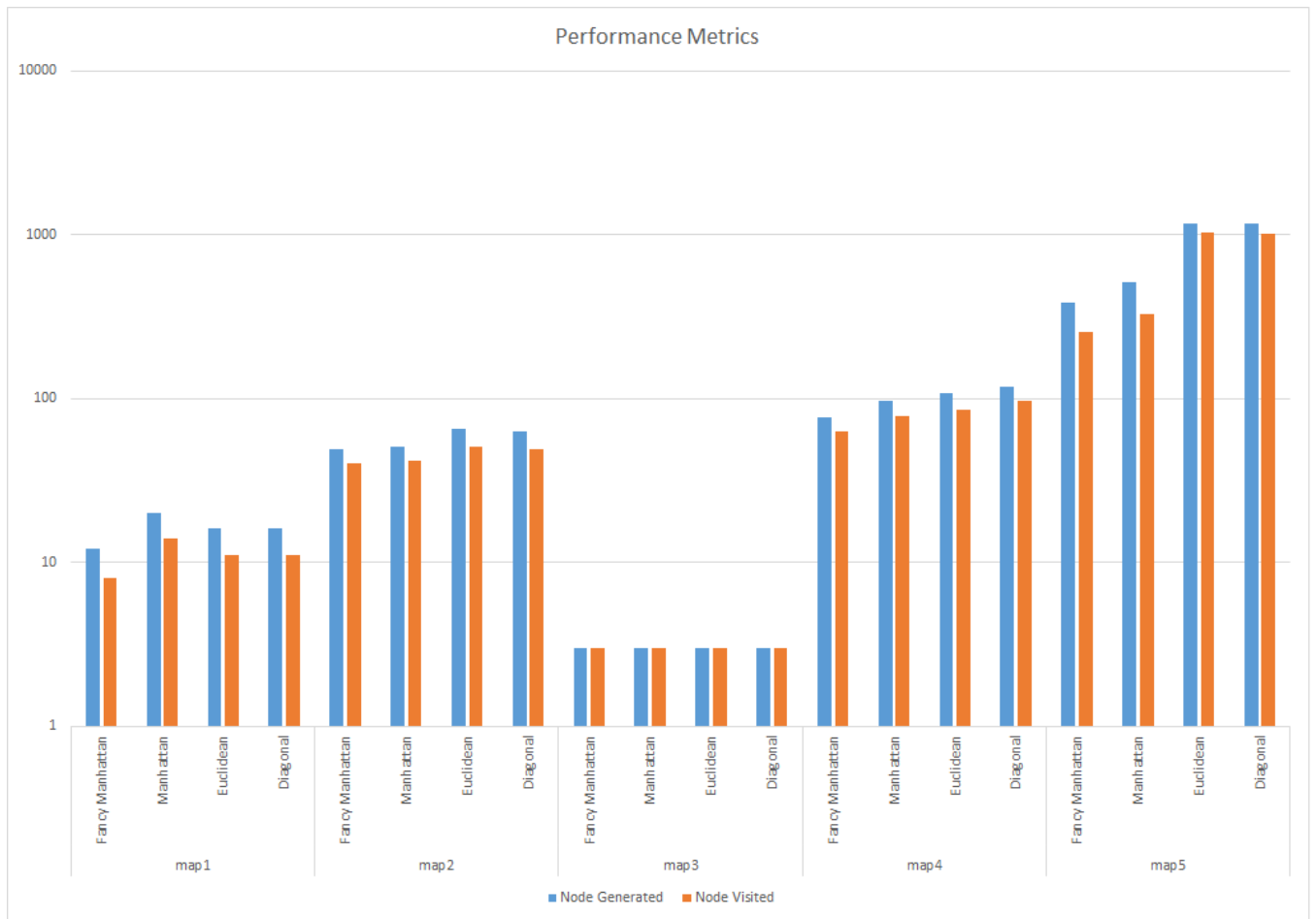Figure 2: Performance of various Heuristics for Map(1-5)

Figure 3: Performance of various Heuristics for Map(1-5)

4   Discussion

After conducting experiments by executing the A* search algorithm for given problem we have found that our suggested fancy Manhattan and Manhattan heuristic have performed better than euclidean and diagonal heuristic. As we discussed in the section 2, Manhattan heuristic is a dominant heuristic over euclidean and diagonal heuristic because:

$$\mathbf{h_{Manhattan}(n) > h_{Euclidean}(n)} \quad \text{and} \quad \mathbf{h_{Manhattan}(n) > h_{Diagonal}(n)}$$

For the given problem the dice is only allowed to move in north, south, east and west direction. Hence in the ideal case dice has to take at least $\mathbf{h_{Manhattan}(S)}$ steps for solving the maze. However, the estimations provided by the Euclidean and Diagonal heuristic on average are much lesser than actual path as they calculate the distances of diagonal paths, which dice is not allowed to take, so these estimations are not near to the reality.

Fancy Manhattan Heuristic has given best result as it reduces the number of nodes explored before reaching to the goal as it gives precedence to the nodes which can take the dice to the goal positions with 1 on the top just by following the path taken by Manhattan Heuristic while calculating estimate.

We came up with many other Heuristics, but either they were not consistent or they were computationally heavy so we didn't use them as they were violating one of the key requirement which has to be a consistent heuristics. Hence a heuristic function apart from being admissible, it shouldnt be computationally heavy else we might end up doing overall more computation than the dijkstra algorithm. i.e., We might end up with expanding less number of states in A* search algorithm, but will result in increasing the time complexity since for calculating heuristics of each state, we would be expanding the states to give better values.

References

[1] Stuart J. Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach*. Pearson Education, 2 edition, 2003. ISBN 0137903952.