A* Project Write-Up

Ben Cohen and Ben Brittain

March 29, 2013

Problem Definition

We define this search problem as attempting to find the optimal path to roll a die through a maze from a pre-defined start to a pre-defined end.

States

We define a state description as a representation of the board with indication of where the start node is, the end node is, and where the die is, as well as information about the orientation of the die. For simplicity, our program outputs state descriptions as a nxm grid of "."s (empty passable nodes),"*"s (empty, unpassable nodes), one "S" signifying the start state, one "G" signifying the goal state, and one number, signifying the die, as well as the number currently on top of it.

Initial State

Our initial state is marked with an "S" in the maze.

Actions

Our set of possible actions is simply defined as rotating our die 90 degrees in each of the cardinal directions.

Transition Function

Our transition function is defined as follows: board[row][col] x rollEast() = board[row][col+1] (provided col is not equal to the number of colums in the board [ie. the rightmost row], in which case rollEast() is an invalid action)

board[row][col] x rollNorth() = board[row+1][col] (provided row is not equal to zero [ie. the top row], in which case rollNorth() is an invalid action)

board[row][col] x rollSouth() = board[row-1][col] (provided row is not equal to the number of rows in the board [ie. the bottom row], in which case rollSouth()

is an invalid action)

board[row][col] x rollWest() = board[row][col-1] (provided col is not equal tozero [ie. the leftmost row], in which case rollEast() is an invalid action)

Goal State

Our Goal state is simply defined as the node marked with a "G" on our board.

Heuristics

For our project we implemented three different heuristics: Manhattan Distance, Direct Cost, and Goal Bias. Below are descriptions of, as well as arguements for the validity of these three heuristics.

Goal Bias

Goal Bias is the most domain specific of the following implemented heuristics. In an A* search, the better the hueristic is at estimating the actual cost, the less unfruitful nodes that are generated. Manhattan Distance (described below) is an excellent way to predict estimated cost in a 4-movement maze. The additional constraint here, which makes Manhattan Distance unsuitable, is the dice rotation constraints. When at the goal, the dice must have a 1 facing up. In addition, the dice can not have a 6 facing up at all during the entire final maze solution. The Goal Bias solution is to weight the columns/rows that are ± 1 from the goal, assuming that the dice can then be rolled into the goal without deviation. The weighting maintains it's admissibility because it never overestimates the cost. Any position outside of this zone is scored by the manhattan distance + 1. Any dice roll outside the zone will take a minimum of the distance in addition to 1-3 more rolls that make the dice line up with the goal. This strongly prunes the generated states in such a way that the number of nodes often decreases by an order of magnitude.

Manhattan Distance

The Manhattan distance is simply the minimum number of moves from the current state to the goal state, disregarding all rules regarding die orientation, as well as walls. This number is guarenteed to be less than or equal to the actual minimum number of moves (ie. following the rules). Consider the below cases:
Case 1: The moves considered without paying attention to the rules are the same moves that would be made while considering the rules. In this case, the two numbers are equal, meaning the heuristic is valid.
Case 2: The number of moves estimated considers an "illegal" move, meaning that in reality more than one legal move would have to be made to get to that state. This means that our estimate will be lower than the actual number, making it an admissable heuristic.

As we have seen, this heuristic essentially makes it an easier problem, meaning that the estimate will always be less than or equal to the actual number.

Direct Cost (Euclidian Distance)

This is perhaps our simplest heuristic. All this does is examine the straight line distance between two points using the pythagorean theorum. This is valid because the shortest distance between two points is a straight line, so this heuristic always provides an optomistic estimation, as the best our die can do is go in a straight line.

Performance

	Goal Bias	Manhattan	Euclidian Distance	Nodes Visited
Puzzle1	7	13	21	6
Puzzle2	43	58	99	16
Puzzle3	BAD	BAD	BAD	BAD
Puzzle4	65	93	158	21
Puzzle5	265	811	4751	26

Discussion

Our results were about as we expected them to turn out. Our "Goal Bias" heursistic was the one we spent the most time crafting, and not surprisingly it performed much better than the rest, especially with Maze 5. Additionally, the Euclidian Distance was the easiest to implement, as well as the simplest, so it's not surprising that it performs rather poorly, especially in a puzzle like this where there are more factors than position (ie. die orientation and walls) to take into account. The Manhattan distance performed alright, and was very close to the "Goal Bias" heuristic in the smaller puzzles.

Another thing which is interesting to note is the convergence between the nodes generated for the Manhattan cost and the goal bias heuristics as the number of walls in the maze went up. We found that the goal bias was most accurate in mostly empty mazes, however peformance slowed down signifigantly when obstacles were added. If we were implementing this to be used on larger mazes, something to consider would be to check the number of walls in a maze, and pick a heuristic based on that. This way, something like goal bias can be applied where it is sucessful, but can be replaced by a different heuristic when it's not as optimal.