Stephen Chambers

09/08/15

Assignment 1 Writeup

**1. What is the size of the state space for this problem?**

State space = ((r\*c - b) \* 5)\* d

Where:

r = the number of rows in the virtual world

c = the number of columns in the virtual world

b = the numebr of blocked cells in the virtual world

d = the number of dirty cells in the virtual world.

Reasoning:

The robot can move to every cell in the world (r \* c), so all of those states need to be added. The robot cannot move to blocked cells, so the amount of blocked cells (b) needs to be subtracted. Now we have a number representing everywhere in the world that the robot can go.

For each of those states, there are 5 possible states that can be generated (moving in all four cardinal diretions plus vaccuming). Ideally, this number should really be the average amount of states that can be generated from a particular state, because most the number of states generated is most likely going to be less than 5. 5 is used for simplicity’s sake. This final number needs to be multiplied by the amount of dirty cells(d), because the whole world is “open” again once we clean up a dirty cell.

**2. Describe any implementation choices you made that you felt were important. Mention anything else we should know when evaluationg your program.**

Depth-first search: Was done with an iterative solution. The open list was a stack, and cycle checking was done linerarlly by having each node keep track of its parent.

Uniform-cost: Open list was a priority queue, sorted by the gscore of the node. For this assignment, the priority queue is no different from a regular queue. The closed list was done with a hash table. My hash function guarantees that if one node had dirty cells (1,2,3) and another node had dirty cells (3,2,1), they would hash to the same value

**3. What is the time and space complexity of each algorithm you implemented? Which algorithms are admissable?**

Depth-first search:

Time complexity- O(