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) )\* 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. 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. The result is the number of **unique** states in vaccum world.

**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(bm)

Space Complexity- O(b\*m)

*Uniform-cost search:*

Time complexity- O(bd)

Space Complexity- O(bd)

Uniform-cost was an admissable algorithm, i.e. it found the optimal solution. Depth-first search was not.

**4. Provide empirical results confirming your answers to the previous question.**

*Depth-first Search*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| World | Time(s) | Nodes Expanded | Nodes Generated | Plan length |
| tiny-1.vw | ~0.00 | 4 | 10 | 4 |
| tiny-2.vw | ~0.00 | 12 | 29 | 11 |
| small-1.vw | ~0.00 | 82 | 241 | 77 |
| hard-1.vw | ~1.00 | 5383 | 13994 | 525 |
| hard-2.vw | Timed out | Timed out | Timed out | Timed out |

*Uniform-cost Search*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| World | Time(s) | Nodes Expanded | Nodes Generated | Plan length |
| tiny-1.vw | ~0.00 | 11 | 15 | 4 |
| tiny-2.vw | ~0.00 | 75 | 113 | 9 |
| small-1.vw | ~0.00 | 899 | 1094 | 24 |
| hard-1.vw | ~1.00 | 2507 | 2577 | 59 |
| hard-2.vw | ~1.00 | 281534 | 300220 | 141 |

**5. What suggestions do you have for improving this assignment in the future?**

I would mention that DFS may not be fast enough to run the more complex problems. I should’ve ran the reference solution against them, but I spent 3 days trying to make my DFS faster when the reference solution couldn’t even solve those problems.

**Depth-first Search Validitor Output**

\*\*\* tiny-1.vw \*\*\*

Program exited with status 0

Solver used 0.000 Seconds

Solver used approximately 3 MB of memory

Plan Cost: 4

Plan Length: 4

Nodes Generated: 10

Nodes Expanded: 4

\*\*\* tiny-2.vw \*\*\*

Program exited with status 0

Solver used 0.000 Seconds

Solver used approximately 3 MB of memory

Plan Cost: 11

Plan Length: 11

Nodes Generated: 29

Nodes Expanded: 12

\*\*\* small-1.vw \*\*\*

Program exited with status 0

Solver used 0.000 Seconds

Solver used approximately 3 MB of memory

Plan Cost: 77

Plan Length: 77

Nodes Generated: 241

Nodes Expanded: 82

\*\*\* hard-1.vw \*\*\*

Program exited with status 0

Solver used 1.00 Seconds

Solver used approximately 3 MB of memory

Plan Cost: 525

Plan Length: 525

Nodes Generated: 13994

Nodes Expanded: 5383

\*\*\* hard-2.vw \*\*\*

Time limit reached

Solver used 5.00 Seconds

Solver used approximately 3 MB of memory

Plan was invalid

**Uniform-cost Search Validitor Output**

\*\*\* tiny-1.vw \*\*\*

Program exited with status 0

Solver used 0.000 Seconds

Solver used approximately 3 MB of memory

Plan Cost: 4

Plan Length: 4

Nodes Generated: 31

Nodes Expanded: 11

\*\*\* tiny-2.vw \*\*\*

Program exited with status 0

Solver used 0.000 Seconds

Solver used approximately 3 MB of memory

Plan Cost: 9

Plan Length: 9

Nodes Generated: 197

Nodes Expanded: 75

\*\*\* small-1.vw \*\*\*

Program exited with status 0

Solver used 0.000 Seconds

Solver used approximately 3 MB of memory

Plan Cost: 24

Plan Length: 24

Nodes Generated: 2730

Nodes Expanded: 899

\*\*\* hard-1.vw \*\*\*

Program exited with status 0

Solver used 1.00 Seconds

Solver used approximately 3 MB of memory

Plan Cost: 59

Plan Length: 59

Nodes Generated: 7251

Nodes Expanded: 2507

\*\*\* hard-2.vw \*\*\*

Program exited with status 0

Solver used 2.00 Seconds

Solver used approximately 3 MB of memory

Plan Cost: 141

Plan Length: 141

Nodes Generated: 796705

Nodes Expanded: 281534