

A1 – CS4300

Random Actions in a Wumpus World

1. Introduction

To being exploring the Wumpus World we set out to discover the how many boards were actually solvable as seeing how well navigating randomly would lead to a successful end. It uses random algorithms to generate both Wumpus World boards as well as a random agent, and the issues to be addressed are:

- How many possible 4x4 solvable boards with a Wumpus, gold, and 0 to 14 pits are solvable?
- How successful would a random agent be in a solvable board?

2. Method

Most algorithms used for calculations are loops that create enough data and store it into a vector. The data contained in these vectors is then used to calculate the mean, variance, confidence intervals, and success percentages needed for the different problems.

The most complex code is in CS4300_A1_Part_II_2.m where it simulates the actions given by CS4300_agent1.m. It is a loop where it always asks the agent for the next step and as long the step doesn't take the agent out of bounds it is taken and then evaluated. If the step resulted in the agent falling into a pit or reaching the gold, then it is the end of the execution. When its execution ends it returns the number of steps and result so those can be used for computations.

3. Verification of Program

The method was applied to boards with 0, 1, and 14 pits 1000 times and the results are found to be:

- 0 pits: 940 were solvable
- 1 pits: 865 were solvable
- 14 pits: 62 were solvable

This is consistent with hand calculations. In the case of 0 pits, there is only 1 possible board that is unsolvable, which is where the Wumpus is places at the agent's entry point. This accounts for 1/16 of the possible boards.

```
>> 1000/16  
ans = 62.5000  
>> 1000-(1000/16)  
ans = 937.5000
```

For 1 pit the unsolvable boards are where the Wumpus or pit are where the agents starts, or if they are blocking the agent into the corner, or blocking the gold into a corner. Wumpus or pit with agent

Gold in corners blocked by Wumpus and pit:

Wumpus and pit blocking in agent:

Unsolvable boards =

Percentage unsolvable =

Out of 1000 trials:

This is exactly what we found by running

```
>> temp = 0;
for i = 1:1000
    if CS4300_Wumpus_solvable(CS4300_gen_board_A1(1))
        temp = temp + 1
    end
end
```

For 14 pits, it is the opposite of 0 pits, where the only solvable board is when the gold is placed when the character starts. From the 0 pit analysis this value should be 1/16 of the boards or about 62, when 1000 random boards were generated, which matched our findings.

4. Data and Analysis

The data for Part I comes from calculations produced by CS4300_A1_Part_I_2.m when computed with 5000 boards per test.

Solvable Boards with 0-14 pits:

- Mean – 1.8055e+03
- Variance – 1.8055e+03
- 95% Confidence Interval – (Mean) ± 635.9227

The data for Part II comes from calculations produced by CS4300_A1_Part_II_2_runner.m when computed with 2000 trials. This test uses CS4300_A1_Part_II_2.m for each simulation. The simulation gets its instructions from CS4300_agent1.m.

- Mean agent steps – 26.5940
- Variance of agent steps – 461.8162
- Successful agent percentage – 1.15%
- 95% Confidence Interval of mean agent steps – (Mean agent steps) ± 0.7905

The data from Part I shows that a large portion of the boards are solvable with a fairly large margin of error because of the many combinations of boards possible. When you

compare this information with the data from Part II which shows that CS4300_agent1.m is not very effective at solving the boards there is you can learn.

Having just random actions results in about 1% of the games having the agent reach the gold. This means that if the agent was to just continue to make random choices for each action step and that it was not only attempting to get the gold but to return to location (1,1) it would only do so 0.01% of the time. This shows the importance of creating agents that are rational and not just random.

5. Interpretation

We found that on average almost exactly half of the boards are solvable (53.74%) which we found to be quite surprising. We expected the value to be lower, and not so close to being half of the boards. We also found that a random agent is NOT a good way to solve the Wumpus boards. With an average success rate of just over 1%, we feel that we would be able to improve on this greatly. However, we were taken aback by how long he could wander around the board without finding the board or dying. The highest number of steps the agent took was almost 180 with many of the trials ending up with 100 - 120 steps taken before dying or finding the gold.

For future projects we would like to be use more of the percepts to be able to create a map of where the agent thinks that pits or the wumpus are to be able to determine the safe places to go and increase success rate.

6. Critique

This lab was effective for introducing MATLAB, the concepts of agents, testing, and beginning an evaluation of the Wumpus World. With the simplified constrains as compared to the normal full set of rules it was able to teach how complex it can become for an agent to make decisions based on the percepts it is given despite the fact that percepts could be completely ignored in this lab.

There were no specific issues or problems in this lab. Any difficulty experienced was from a lack of knowledge about MATLAB and the need to review some basic probability and statistics equations.

7. Log

Braden Scothern

- 60 min — Part I code
- 60 min — Part II code
- 45 min — Lab Report — Sections 2, 4, and 6

Kyle Heaton

- 45 min — Random agent
- 45 min — Confidence intervals and corroborating code
- 90 min — Lab Report — Sections 1, 3, and 5

Appendix

MATLAB Code Files with brief description:

- CS4300_agent1.m – A simple agent for Wumpus World. It selects randomly from the actions: MOVE_FORWARD, TURN_RIGHT, and TURN_LEFT
- CS4300_A1_Part_I_1.m – A function that takes a number of boards to generate and the number of pits for each board. It then returns how many of the randomly generated boards are solvable.
- CS4300_A1_Part_I_2.m – Take a number of boards to generate for each test and tests that many boards across the entire interval of 0-14 pits. It then returns the mean number of solvable boards, the variance of solvable boards, and the 95% confidence interval.
- CS4300_A1_Part_II_2.m – This will use CS4300_agent1.m to get instructions and use it to simulate a game of Wumpus. It is very similar in train of thought to the given CS4300_run_A1_trials.m but differs in the fact that it returns the number of steps taken and the result of the simulation.
- CS4300_A1_Part_II_2_runner.m – Takes as input a number of times to use CS4300_A1_Part_II_2.m to simulate games. It then calculates the mean steps, variance of steps, success rate, and 95% confidence interval for the mean.