A5 – CS4300 Assignment A5 Lab Report

Monte Carlo Probabilistic Agent

By Haochen Zhang & Tim Wei
11/08/2017

1. Introduction

In this assignment, we study about the effect of a probabilistic agent. *CS4300_MC_agent* is the function written for this. It uses *CS4300_WP_estimates*, a function that use Monte Carlo to estimate the probability of W and P in each cell, to determine its path. We are interested to answer the following question:

- How does the sample size of MC impact the performance of *CS4300_MC_agent*?

2. Method

The framework of *CS4300_MC_agent* is based on a previous assignment agent, *CS4300_hybrid_agent*, instead of using a knowledge and RTP to generate agent's action, we use *CS4300_WP_estimates* which apply Monte Carlo methods to update pit and Wumpus likelihoods in the Wumpus world.

Everything else is the same for *CS4300_MC_agent* compare to *CS4300_hybrid_agent* except for the following:

- Find a safe cell
- Locate the position of the Wumpus and shoot
- Find a OK cell
- Die trying
- Handling Scream precept

We have a 4 x 4 matrix called *danger_P* which is the addition of *pits_P* and *Wumpus_P* given by *CS4300_WP_estimates*. In order to find a safe cell to go, our agent will find the cell that *danger_P* is 0 at that position. If multiple cells are safe, our agent will choose the closest cell. To locate the position of the Wumpus, our agent will check *Wumpus_P* and find which cell has the probability of Wumpus existing larger or equal to 50%. The shooting sequence are the same as *CS4300_hybrid_agent.* While try to find a OK cell to go, our agent will find the cell will lowest *danger_P*. If there is a tie, it will go to the closest cell. Unlike *CS4300_hybrid_agent* which will climb out if will cells are dangerous, *CS4300_MC_agent* will die trying. If agent receive Scream, it will make *Wumpus_P* to be 4 x 4 matrix filled with zeros every time after it has been updated by *CS4300_WP_estimates.*

For *CS4300_WP_estimates*, we keep generating boards and check if it matches our percept(*breezes* and *stench*). Once it matches, we increment our counter by 1 and once it reaches the given number of trial, it will stop and return *pits_P* and *Wumpus_P*. While generating boards, we have a maximum board count that is way larger than number of trials to constrain the total number of boards. If it reaches our limit (say 10000 boards) and have at least one board generated, it will return the result it get so far and have a flag called *fail* set to 1.

Then, we use 50, 100 and 200 MC samples to determine a mean score for each one of the given 250 test boards in terms of a mean, variance and confidence interval. We also record number of successes and number of failures for those boards. Comparing these to the same agent without MC(setting the trials to be 0), we can determine what impact MC has on scores and have the results shown in a table.

3. Verification of Program

The method is to give *CS4300_WP_estimates* such percepts so it can only return a specific possibility set. The following percepts are set up so that $\neg P_{21}$, $\neg P_{41}$, $\neg P_{32}$, P_{12}, and W_{21}.

```
>>  breezes = -ones(4,4);
stench = -ones(4,4);
breezes(4,1) = 1;
breezes(4,3) = 0;
stench(4,1) = 1;
[pts,Wumpus,fail] = CS4300_WP_estimates(breezes,stench,50)
pts =
   0.2200   0.2000   0.1800   0.2000
   0.2200   0.1800   0.2000   0.3000
   1.0000   0.1600   0        0.2400
   0        0        0.2400   0
Wumpus =
    0    0    0    0
    0    0    0    0
    0    0    0    0
    0    1    0    0
fail =
    0
```

4. Data and Analysis

The following table shows the data of the agent's performance using 0, 50, 100 and 200 MC samples in terms of a mean, variance and confidence interval. We also record number of successes and number of failures for those boards. The result shows that using MC will significantly increase the score of the agent. It will also decrease its failures.

	Mean Score (±95% CI)	variance	Number Successes	Number Failures
No MC	-478.6240 (±109.1178)	774850	66	184
50 Samples	395.7240 (±112.1841)	819010	178	72
100 Samples	388.5480 (±112.5351)	824140	177	73
200 Samples	364.2880 (±113.8975)	844220	174	76

5. Interpretation

In this assignment, we set out to answer the question:

- How does the sample size of MC impact the performance of *CS4300_MC_agent*?

From the data we got, it seems like there is only significant impact when the agent runs with or without MC. The performance even seems to drop in all aspect as sample size increase. This may be caused by chance though, since success on some board rely purely on luck. The reason that sample sizes' impact is low can be that during the first few actions, the percepts are not complete enough to make a difference, and during the last actions, only few boards would fit the percepts received so the generated board count exceeds the maximum number easily.

It might be helpful for this study to look into lower sample sizes. There may be a threshold after which sample sizes make less impact, since during the debug phase of this assignment, too low of a sample size proved to be a problem.

6. Critique

While doing the statistic part of this lab, we found some undetected bugs that cost us some time to fix them, not to mention starting the program all over again. Since we only tested our agent on one complex board, there are still some corner and boundary cases that had not been covered. Doing more tests before writing this report will be useful to solve this issue.

Another thing that we are concerned about is the maximum number of boards being generated. We just set a large enough number to constrain the runtime. The reasonable thing is to time how long it will take for each different maximum numbers before deciding them. It would not cost us as long as we waited for our statistic to come out.

7. Log

Haochen Zhang(Section 2, 4, 6)
A total of 6 hours was spent performing the experiment in Matlab.
A total of 6 hours was spent performing the experiment in writing the report.

Tim Wei (Section 1, 3, 5)
A total of 6 hours was spent performing the experiment in Matlab.
A total of 6 hours was spent performing the experiment in writing the report.