Dusty Argyle (1,2,3,4,5,6,7)
Scott Hoge (2,4,6,7)
November 1, 2016
CS 4300 - Artificial Intelligence

A5

1. Introduction

Monte Carlo is a clever algorithm for generating probabilities. In this assignment, it is used to make educated estimations for the Wumpus World agent. This method works by having information kept by the agent involving the percepts. The stenches and breezes help determine safe places by eliminating randomly generated boards that do not fit with the board configurations.

While Monte Carlo is very useful, it does require a significant amount of samples to be consistent and accurate. This can be proven by asking the following questions: Does the Monte Carlo algorithm become more accurate as more samples are added? Does the agent score more with more samples? How does the agent react with no samples?

2. Method

The method of Monte Carlo is conducted by our Wumpus World agent. This agent calls for an estimation every time a percept changes on the board. So if the agent moves into 2,1 and there is no breeze or stench, this can be saved and used to more accurately estimate solutions.

When the agent requests for a re-estimation based on it's new data, it passes with it all of the places that the agent has received percepts. In our method, we made a function that would determine places that would not work with our current perceptions. We would generate a random board with a .2 probability of having pits and check it against these places. If it fit the description presented by the perceptions, the board would be counted.

To count the boards we would determine the key locations of things in the board that was generated. For example, if the board generated a Wumpus in (2,1) and it passed our tests, we would count that as a possible location for the Wumpus and later on we would divide all of the counts by the number of samples to get the percent chance that a Wumpus or Pit was in the cells. This gets dangerous as the boards are not perfectly distributed and some cells my generate 0 percent chance of obstacle when really, there just wasn't enough samples to get an obstacle in that location. The number of samples is a key factor in Monte Carlo's execution and accuracy. The agent might make decisions based on these probabilities with its life on the line.

3. Verification of Program

I did confirmation by watching the traces as they would go through the boards given for us to collect data on. I also did the boards by hand to show that the boards were solvable. Since I used the skeleton provided in class, the agent does not go forward first. Ideally, this does not matter. The agent should be able to find the “leaks”. I also ran the MC agent on the same tests as what I used last assignment with the logical agent. See below for hand verifications and actual boards.

4. Data and Analysis

The following table demonstrates an answer to our questions asked. As the number of sample sizes increases, the more accurate our agent’s predictions become. There were two boards I included in the report to show how the agent made those decisions. The estimations would get more accurate and therefore the success of the agent became apparent.

	Mean Score	Number Successes	Number Failures
Samples none	-578.5400	60	190
Samples 25	101.3120	143	107
Samples 50	117.0760	145	105
Samples 100	124.9440	146	104
Samples 200	127.9114	146	104

There is a significant trade off when dealing with the number of samples. This can get very expensive in our program when the board has been narrowed down to a specific point. This could cause the random board generator to run for an immense amount of time generating boards that are difficult to generate. Therefore while it is more accurate to have a lot of samples, the run time increase significantly and will reduce performance. It is good to find a good middle ground between the two variables where accuracy may have minimal gains in increasing the sample size.

5. Interpretation

As we can see from the results table, as samples go up, so does the scores and successes. We can also see that the curve projected by the number of samples, begins to flatten as the number of samples increases. The flattening happens for both the successes and score. We can also

see if that when using 0 samples, the agent is almost random. It is slightly smarter than that in that it can determine clear paths, but if any sensory happens, the agent is not smart enough to narrow down potentials.

In the future, it would be better to do longer and larger tests, for example, a significant amount more could be added to the sample size to increase accuracy. This was difficult because of the deadline, so time was a large factor in being able to produce data.

6. Critique

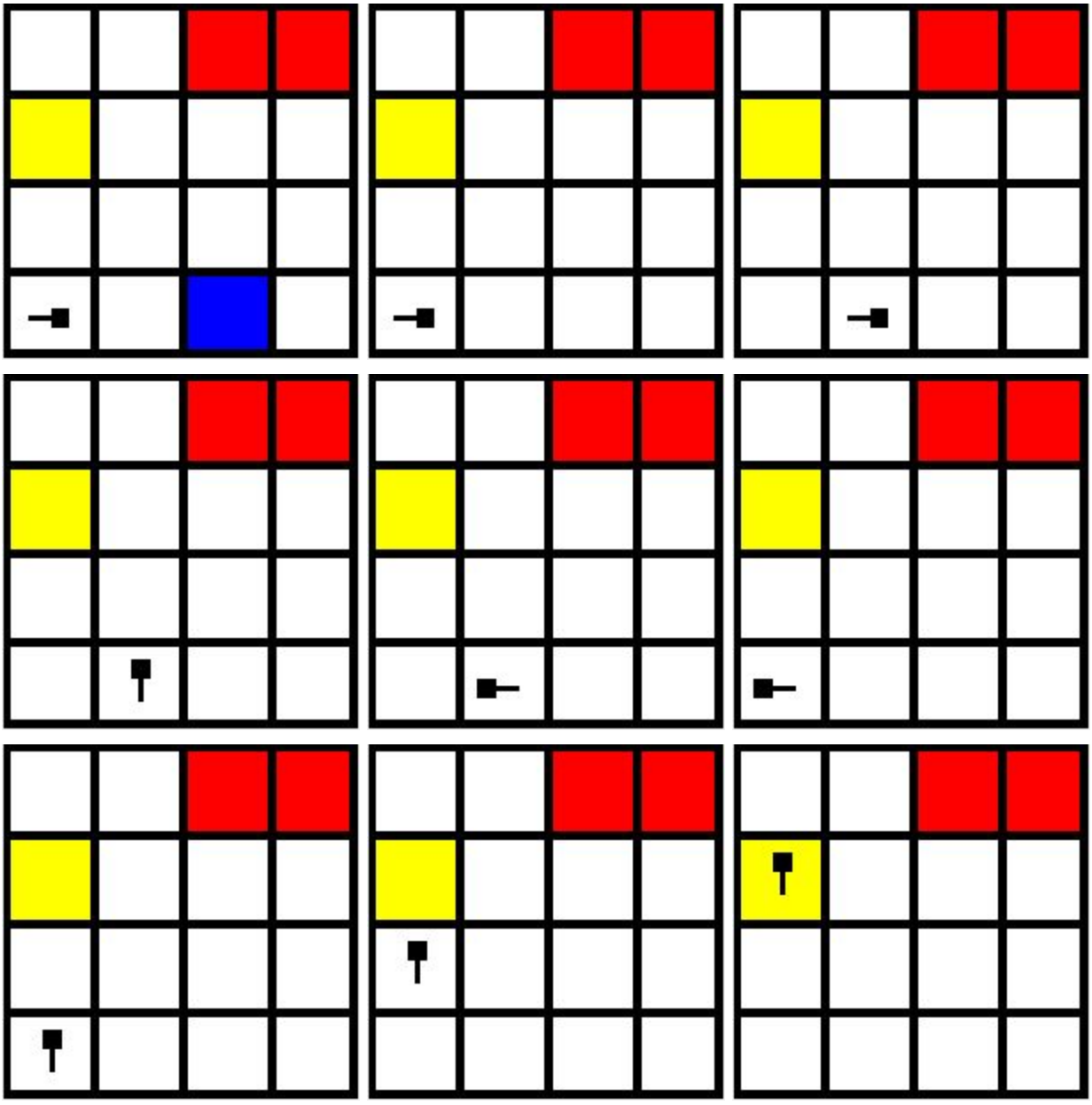
Some important aspects of this experiment include gaining a greater understanding about Monte Carlo as an algorithm and how to use it properly. If we use very little to no sample in our agent, we will see a rapid decline in our success. I gained a greater understanding of this when the tests would demonstrate this very clearly. As my sample sizes would go up, the better my agent would perform in Wumpus World.

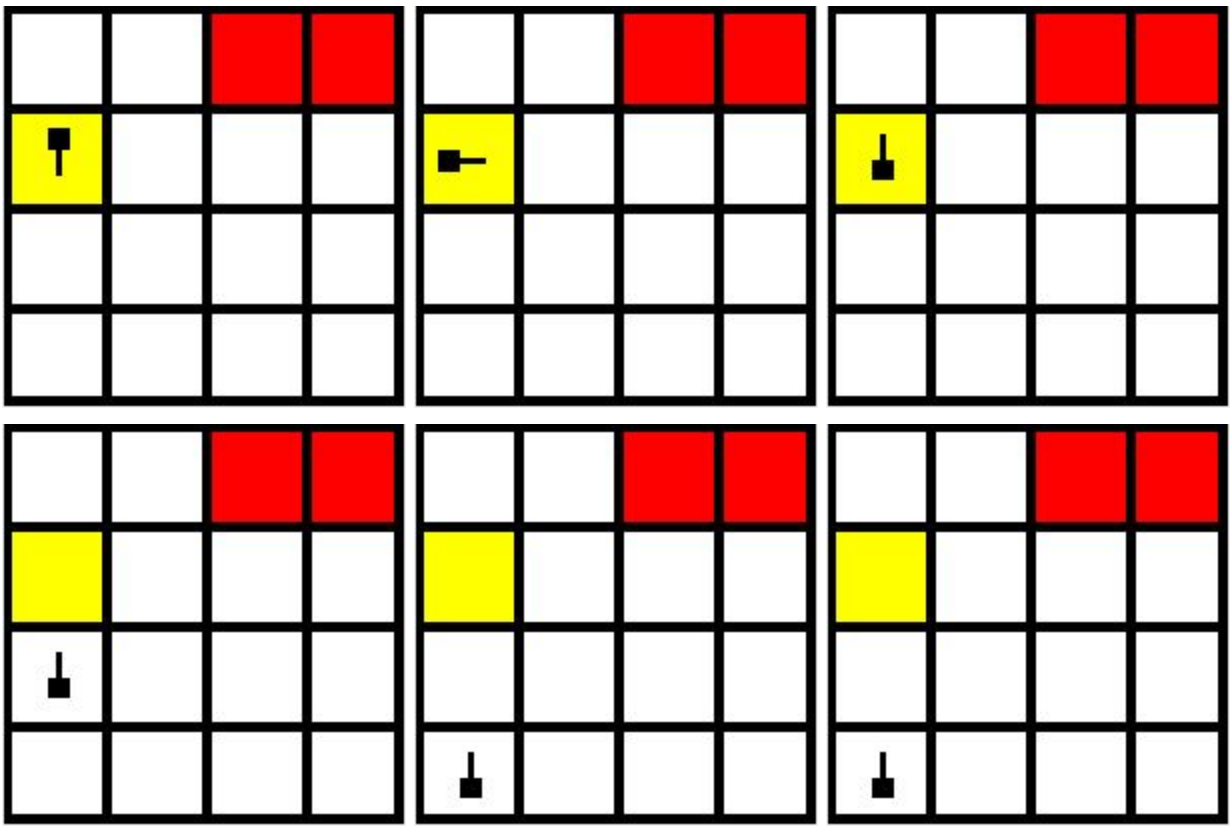
7. Log

Dusty: 15 hours Algorithm, 6 hours report
Scott: 2 hours Algorithm, 0 hours report

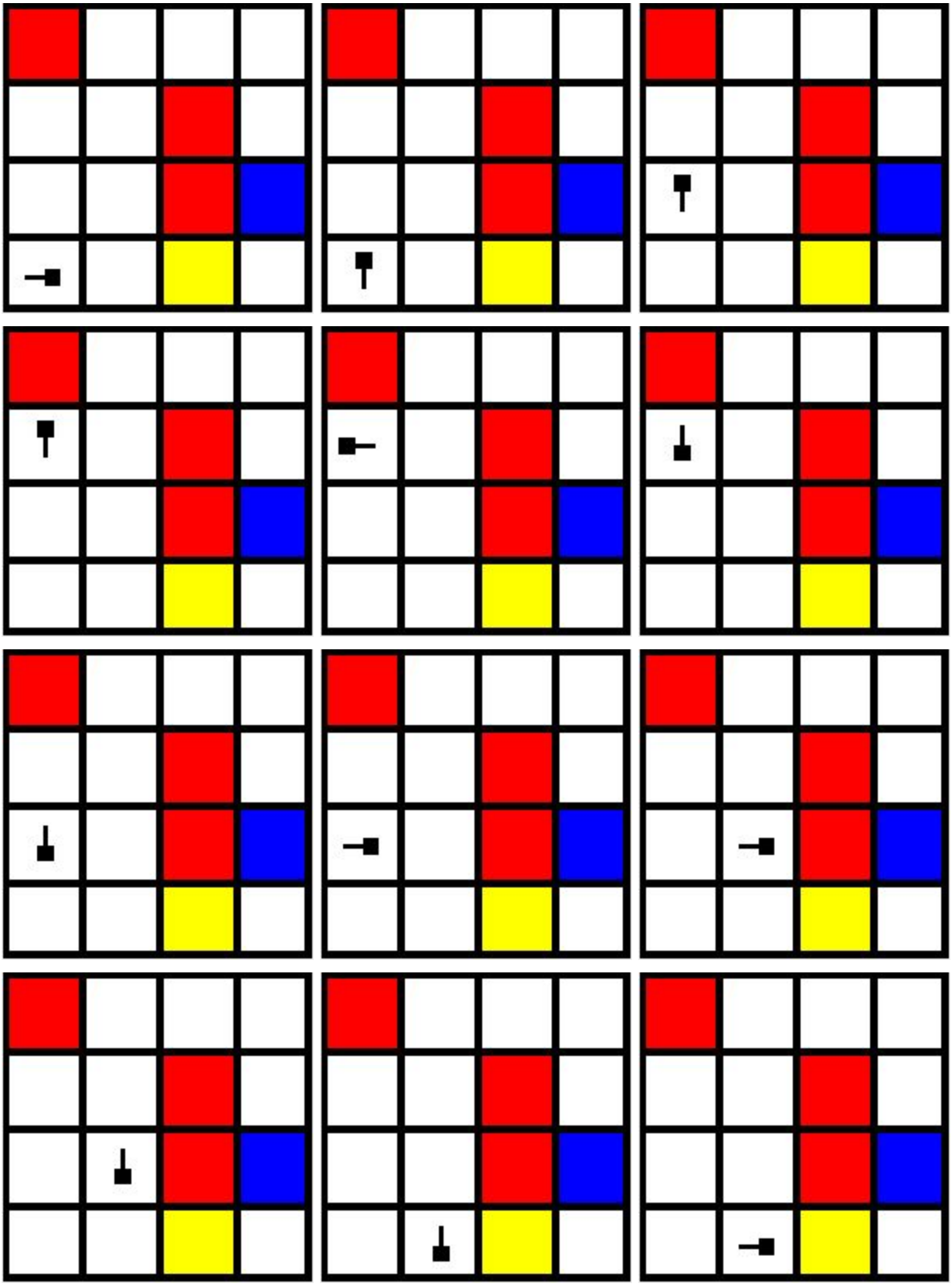
8. Appendix
Verification:

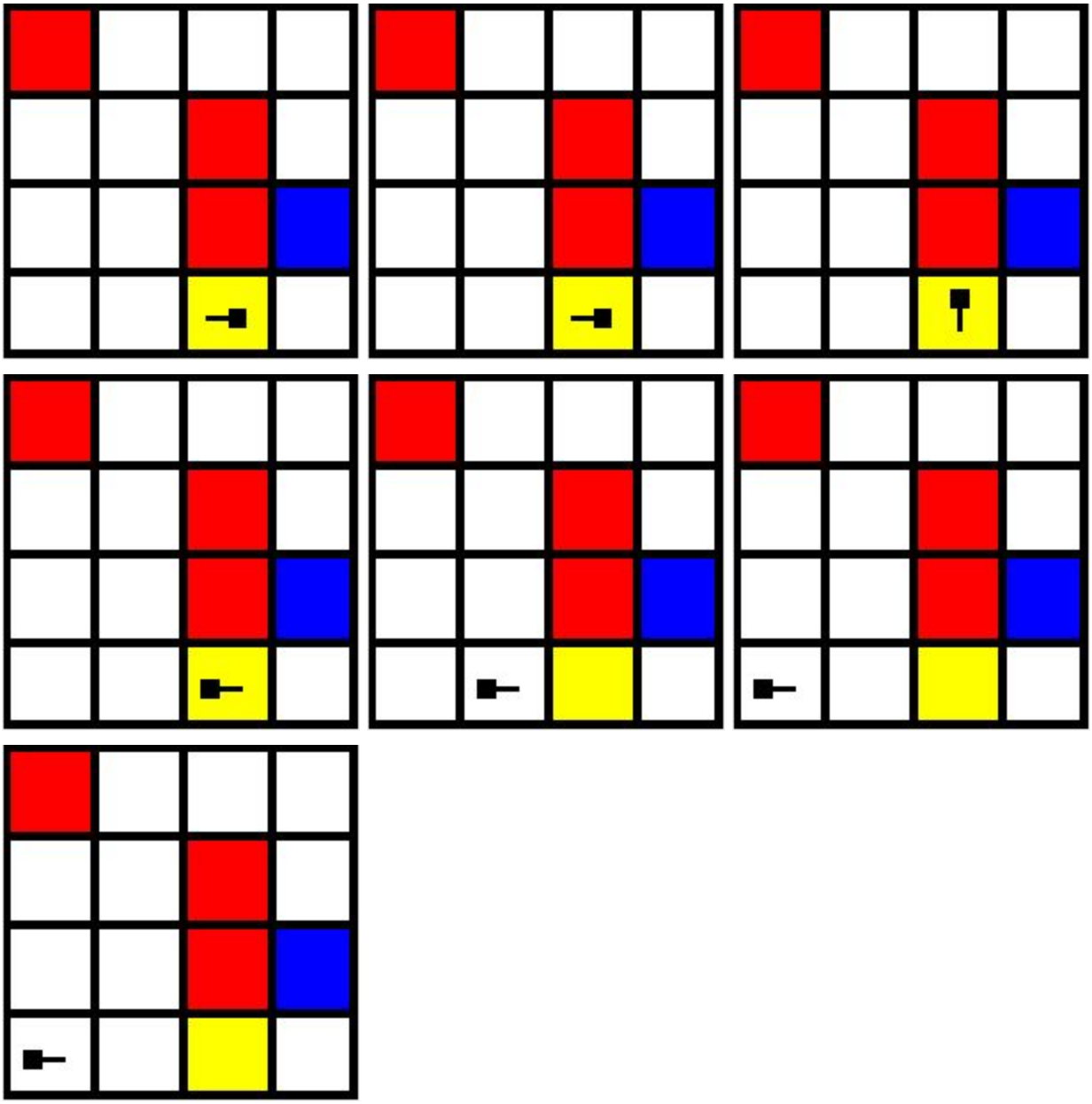
-> Board1:





-> Board2





Verification: