

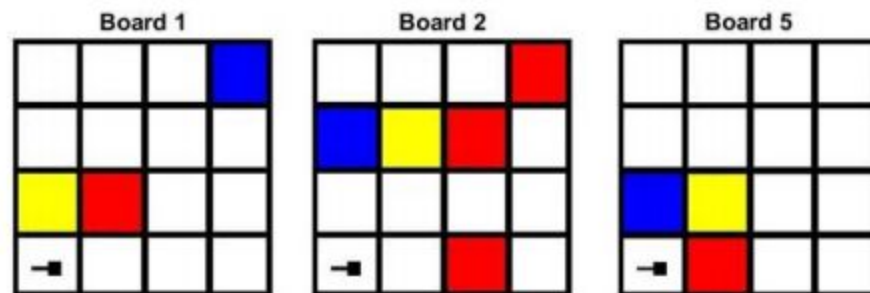
Lab Report A4B  
 CS4300  
 Derek Heldt-Werle 2,4,6  
 Matthew Lemon 1,3,5

10/20/16

## 1. Introduction

For this lab we created a Hybrid-Wumpus Agent function which uses a persistent Knowledge Base and understanding of its world to navigate its way to the gold and then make its way back out. The function also incorporates the Tell and Ask functions using propositional logic and resolution theorem proving to update its KB and garner a greater understanding of its surroundings. It also makes use of A\* path planning to map routes that it can take through the Wumpus World Board.

We used these three boards to test the Hybrid-Wumpus Agent's performance and behavior.



We will pose the following question and answer through testing;

What are the average scores of the Hybrid-WumpusAgent on the 3 separate boards?

## 2. Method

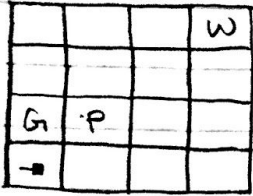
For this lab we were heavily influenced by the wumpus agent hybrid described in Peter Norvig and Stuart J. Russell's book Artificial Intelligence: A Modern Approach. Through the creation of persistent variables that keep track of the agent's current knowledge of pits on the board, where the wumpus is on the board, what the agent knows about the board, the places we have been to, the plan to get to the current goal, and our knowledge base. With these, the agent is able to create and execute a plan to find the gold, grab it, and return safely if possible. We begin by filling our knowledge base with the known rules about pits, breezes, stenches, and wumpus'. After this we are able to begin the execution of our hybrid agent.

The agent begins by asking the knowledge base if there is a gold at our current location. If there exists gold here, we create a plan that grabs the gold, finds a route back to the entrance using

A\*, and escapes. If we don't have a plan we will then update the boards (pits, wumpus, board) with information we have obtained through the percept by calling RTP on whether or not there is a pit or wumpus in a surrounding location for our knowledge base. From here we will figure out a plan based on the safe areas the agent could travel to. If there are new spaces for the agent to go to, we will create the plan using the A\* algorithm to get to that location. If there are no new spaces, we will look at the unvisited nodes we have in the frontier, choose one at random, shoot into it, mark it as safe, then create a plan to get to the now potentially safe space. The agent will then continue to execute the plan until it is yet again empty, and will thus repeat the same process as described.

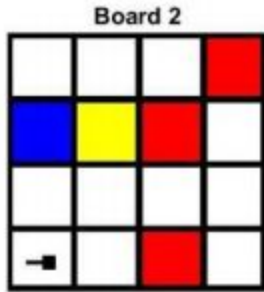
### 3. Verification of Program

#### Board 1:

board 1					
					
Agent	KB	Places	visited	Goal	Plan
Step 0: [1,1,0]	~S <sub>11</sub> ~B <sub>11</sub> ~G <sub>11</sub>	[1,2] [2,1]	[1,1]	[1,2]	[3,1]
Step 1: [1,1,1]	.			[1,2]	[1]
Step 2: [1,2,1]	~S <sub>12</sub> B <sub>12</sub> G <sub>12</sub>		[1,2]	[1,1]	[4,3,3,1,6]
Step 3: [1,2,1]				[1,1]	[3,3,1,6]
Step 4: [1,2,2]				[1,1]	[3,1,6]
Step 5: [1,2,3]				[1,1]	[1,6]
Step 6: [1,1,3]				[1,1]	[6]
Step 7: Climbed Out					
Score = Gold - 7steps = 993 points					

Running the Hybrid-Wumpus Agent produces the same score

#### Board 2:



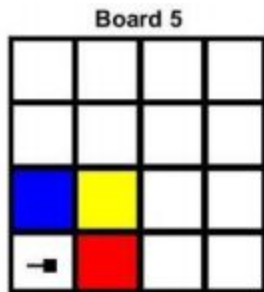
	Agent	KB	Places	Visited	Goal	Plan
Step 0	[1,1,0]	~S11 ~B11 ~G11	[1,2] [2,1]	[1,1]	[1,2]	[3,1]
Step 1	[1,1,1]				[1,2]	[1]
Step 2	[1,2,1]	S12 ~B12 ~G12		[1,2]	[2,1]	[3,3,1,3,1]
Step 3	[1,2,2]				[2,1]	[3,1,3,1]
Step 4	[1,2,3]				[2,1]	[1,3,1]
Step 5	[1,1,3]				[2,1]	[3,1]
Step 6	[1,1,0]				[2,1]	[1]
Step 7	[2,1,0]	~S21 B21 ~G21	[2,2]	[2,1]	[2,2]	[3,1]
Step 8	[2,1,1]				[2,2]	[1]
Step 9	[2,2,1]	~S22 ~B22 ~G22	[2,3] [3,2]	[2,2]	[2,3]	[1]
Step 10	[2,3,1]	S23 B23 G23		[2,3]	[1,1]	[4,3,3,1,1,2,1,6]
Step 11	[2,3,1]				[1,1]	[3,3,1,1,2,1,6]
Step 12	[2,3,2]				[1,1]	[3,1,1,2,1,6]

Step 13	[2,3,3]				[1,1]	[1,1,2,1,6]
Step 14	[2,2,3]				[1,1]	[1,2,1,6]
Step 15	[2,1,3]				[1,1]	[2,1,6]
Step 16	[2,1,2]				[1,1]	[1,6]
Step 17	[1,1,2]				[1,1]	[6]
Step 18	Climb Out					

Score = Gold - 18 steps = 982 points

Running the Hybrid-Wumpus Agent produces the same score

### Board 3:



	Agent	KB	Places	Visited	Goal	Plan
Step 0	[1,1,0]	S11 B11 ~G11		[1,1]		[5]
Step 1	[1,1,0]		[2,1]		[2,1]	[1]
Step 2	[2,1,0]					dead

Score = Died + Shot Arrow - 1 step = -1051 points

Running the Hybrid-WumpusAgent will sometimes choose this course of action which produces the same score.

## 4. Data and Analysis

Board 1:

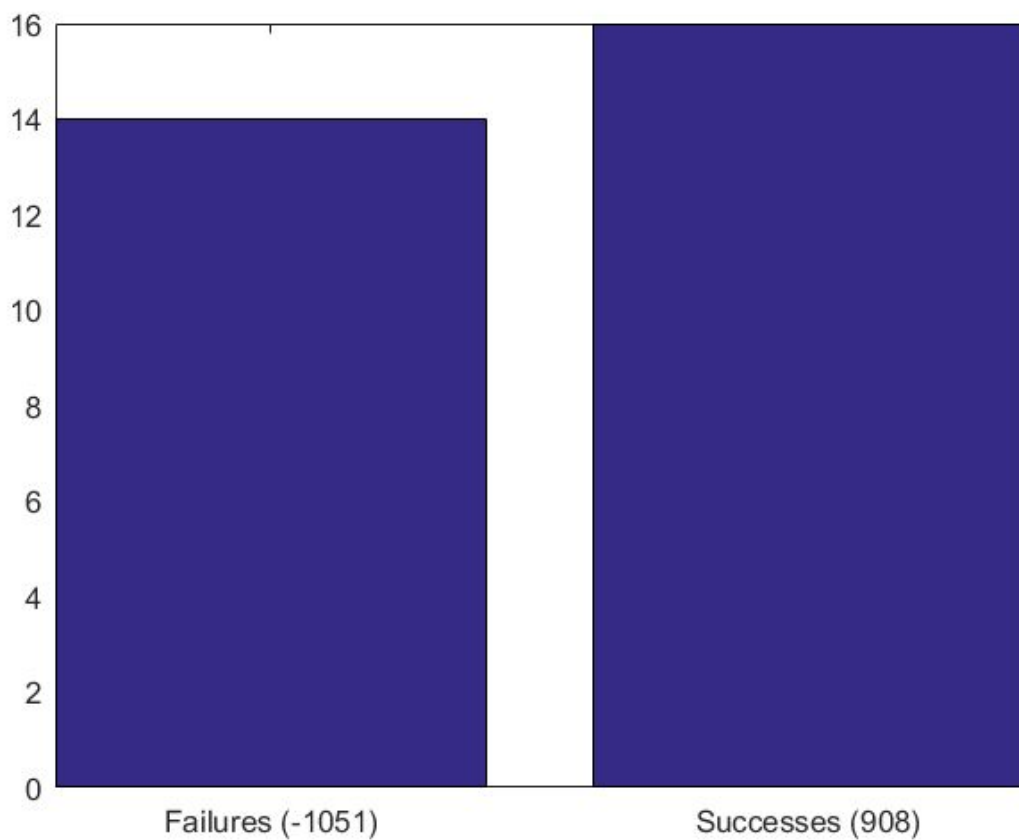
For every run of Board 1 the score will be 993 as described above.

Board 2:

For every iteration of Board 2 the score will be 982

Board 3:

Due to our implementation approach of if we have nowhere safe to go we randomly select a space in our frontier, shoot an arrow into it, then go to that space. For the case in which our agent chooses the [2,1] space to shoot and move into the score will be -1051. If the agent randomly guesses where the wumpus was correctly, the score will be 909 as our depth first search will take a highly inefficient route to get to the gold at 2,2.



**Figure 1: Number of Successes vs. Failures for Board 3 (30 Trials) Big old donkey dicks**

## 5. Interpretation

Because of the logical order in which the Hybrid-WumpusAgent solves the boards when there is a solvable board which does not require the killing of a wumpus it will solve that board the same way every time. This is why the scores for boards 1 and 2 are always the same and as was

shown in the Verification steps it falls in line with how the board would be solved manually as well.

Board 3 presents an interesting challenge in that we do not know where the Wumpus is and there is no safe place to go to. Our Hybrid-WumpusAgent therefore finds a random node on the frontier and shoots an arrow into it and moves into that square in the hopes of finding a solution. For this specific board because there are only 2 places in which we can do this action we have a 50% chance of successfully killing the wumpus and getting the gold. As expected the mean score that the Hybrid-WumpusAgent gets is close to 0, minus 1000 points for dying and plus 1000 points for getting the gold. This can be seen in Figure 1

## 6. Critique

During this assignment we spent a great deal of time furthering our knowledge of boolean algebra and resolution theorem proving. By having to keep track of and maintain the accuracy of the knowledge base, we were forced into having a complete understanding of the problem presented. We also learned about how to work with persistent variables as we kept track of the knowledge we had about the board, pits, and wumpus. This assignment did a great job in encapsulating the previous assignments and compounding of the knowledge gained from them.

## 7. Log

Author Matthew Lemon & Derek Heldt-Werle

Coding Portion (Worked together): 18

Report (Derek): 2

Report (Matt): 2