

# Modelling a player's logical actions through the game "Hunt The Wumpus"

0921741

January 30, 2013

## Abstract

The aim of this report is to give an introduction to the "Hunt The Wumpus" game and discuss observed results gained from modelling the game in JS-Eden after studying a model already implemented in Eden by G. Cole [1]. The game is studied for Artificial Intelligence purposes as its complexity combined with its set environment make modelling the player's approach very interesting. The game environment consists of a static game map, but the player has little knowledge about the other agents in the game.

## 1 Introduction

Empirical Modelling focuses on the development of models representing situations, procedures, events or ideas that take place within an environment and are based on a set of observables that are connected through a set of dependencies. Through these relationships, we are able to model how changes in some observables influence others. Moreover, EM attempts to model representation of how changes in the environment also affect the actions of agents. The models are taken from a wide range of research areas, from models assisting with educational technologies, such as learning processes [4], to models of real-life events and situations such as a railway accident or the operation of a machine that is dependent on the actions of an agent [5].

## 2 Empirical Modelling and Agent Based environments

One of the areas Empirical Modelling is focusing is Artificial Intelligence in the context of

modelling the logical actions that would lead to the achievement of solving a problem. This is achieved through modelling games that are based on solving a problem through logical steps, or modelling multi-agent environments in order to study the actions and strategies of an agent's attempt to achieve a specific goal and how they are influenced by the actions of other agents, changes in the environment or the amount of knowledge the agent has of the system which changes as the agent explores the environment and learns more about the environment's observables. In this paper we will be studying the agent's actions in the context of the "Hunt The Wumpus" game, focusing on the thought process of the player before solving the task at hand at each stage of the game.

## 3 "Hunt The Wumpus"

The "Hunt The Wumpus" game was first introduced by Gregory Yob [3] and it consists of a grid in the shape of a kite-shaped dodecahedron that represents the Wumpus's cave. The edges connecting the 20 nodes represent

the paths between the nodes which represent the rooms. The goal of the game is to kill the Wumpus by shooting an arrow in its room, but to achieve that the player has to overcome other obstacles, such as the danger of falling into a pit or getting moved to a different room by bats. If the player ends up in the same room as the Wumpus, the Wumpus will eat them and the game is over. There is no limit on the number of moves the player has to solve the problem in, but there are only 5 arrows available.

Even though the environment of the game is set from the start and is not subject to any changes (unless playing a version of the game where the Wumpus is not set but can move if you attempt to shoot it and miss), the player has very limited knowledge to start with apart from its initial location. Further knowledge of the environment can be acquired by exploration of the game map, but with the risk of landing in the same room as one of the game's obstacles, due to the fact that the knowledge of the player will often still not be enough to cover all the available paths the player could take. However, as the game has to be solved in a finite number of moves, the player usually has to either determine if a room is safe by eliminating all other possibilities, or take some risks that could potentially lead into losing the game but if not, they will enable them to reach the goal in a smaller number of moves and before running out of arrows.

This game can be solved by a knowledge-based agent that considers all the available actions after each iteration of the game and acts by choosing the action that would be the most profitable in respect to its chosen strategy and that would assist in achieving the game goal. Aside from a strategy enabling the agent to act towards winning the game, the

agent will have to choose another strategy as to how to handle situations where there is not enough knowledge to make a safe move. Through implementing a model in JS-Eden, I have attempted to address this problem by calculating a agent's next move if they were following one of two strategies: taking risks and attempting to solve the problem faster or choosing the safest move and acquiring as much information as possible when there isn't one. This information is calculated after each move and is displayed to the user. However, the player can choose whether to continue by following the move displayed or not, as neither of strategies alone is guaranteed to win, but rather a combination of them would be more ideal.

First, a strategy that prioritises the safety of the player, calculates the next move by always suggesting the safest option forward and if one doesn't exist, it suggests moving back to the previous room and choosing a different option. If the player can "smell the wumpus" from their current room, it shoots arrows in all the possible rooms before moving there. This approach ensures that the player does not lose from falling into a pit or getting eaten by the Wumpus, but in some cases the player might be moving in loops or run out of arrows. The second approach, still calculates the move that will be safest but attempts to get nearer to completing the goal, ie killing the wumpus. For example, in the case where the player can smell the wumpus but returning to the previous room will not assist in winning the game, the player chooses to move forward with the risk of getting eaten. In most cases that does not happen though because usually the player will have an idea of where obstacles are, from detecting them when coming from the opposite direction, and if that is not the case, that strategy is prioritised before taking a risk.

## 4 "Hunt The Wumpus" game model by Cole

G. Cole implemented a model of the "Hunt The Wumpus" game in the Eden environment in 2005. His approach into modelling the player's approach of the game, was the addition of a table of inferences which listed all the assumptions that can be made for each room according to the knowledge acquainted after each move. Each move would add more information about the environment and any obstacles would be instantly eliminated or confirmed for any rooms that there is enough knowledge about. Even though the table on inferences is an excellent way of keeping track of information found about the game observables, the player is not encouraged to follow the same logic procedure for choosing their next move, as it is easier to eliminate possible actions by just looking at the table for the specific room, rather than choosing an action by making inferences after each move.

	WUMPUS	PITS	BATS	PASSAGES
1w7	yes	no	no	[2,5,6]
2wb	no	no	no	[1,3,8]
3w	no	no	no	[2,4,10]
4w	no	no	no	[3,5,12]
5wp	no	no	no	[1,4,14]
6	no	no	no	[1,7,15]
7	no	no	no	[6,8,16]
8w8	no	no	yes	[2,7,9]
9b	no	no	no	[8,10,17]
10	no	no	no	[3,9,11]
11	no	no	no	[10,12,18]
12	no	no	no	[4,11,13]
13	no	no	no	[12,14,19]
14	no	yes	no	[5,13,15]
15	no	maybe	maybe	[6,14,20]
16	no	no	no	[7,17,20]
17	no	no	no	[9,16,18]
18b	no	no	no	[11,17,19]
19	no	no	yes	[13,18,20]
20	no	maybe	maybe	[15,16,19]

Figure 1: Table of inferences in the Cole Model

## 5 Game implementations

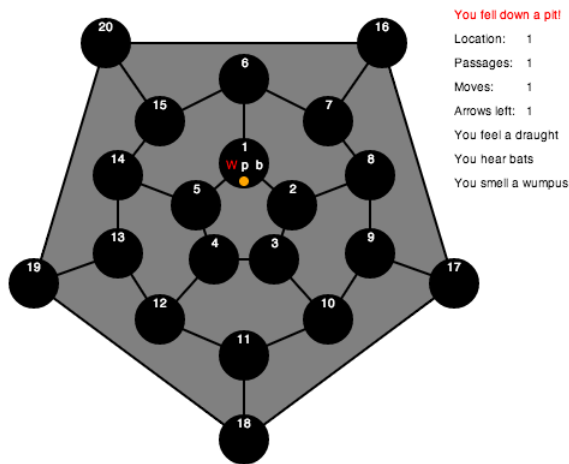
### 5.1 Cole's implementation in Eden

For implementing the model in Eden, Cole used %donald, %eden and %scout notation for drawing the game environment and connecting the changes on the map and inferences table to the game observables that are subject to change value after each move. The game functions and procedures were developed with %eden. Using the %donald notation, Cole was able to treat the game observables as sets of observables compiling the whole environment, which made it easier to define the dependencies between them.

### 5.2 Implementing the game model in JS-Eden

My attempt into implementing the model in JS-Eden [2] is similar to Cole's implementation in respect to the parts developed with %eden notation. However, due to the fact that the rest of the notations aren't available, I developed the game map and using JS-Eden picture functions such as Circle and Line. As a result, each component of the game map can be treated as an observable without having to translate it from the map to actual values. This way, each move directly influences the observables in the model that are depended on the user's actions.

In order to develop a JS-Eden implementation of the game, I first explored the existing model by Cole, in an attempt to determine the observables that would need to be included. I then proceeded to first draw the map of the game by starting with the shape of a 5-sided polygon which resembles the kite-shape of the game's grid. I then completed the map by drawing circles representing the rooms and lines representing the paths. Each circle is drawn based on one specific point on the map,



**You fell down a pit!**  
 Location: 1  
 Passages: 1  
 Moves: 1  
 Arrows left: 1  
 You feel a draught  
 You hear bats  
 You smell a wumpus

**You were eaten by the wumpus!**

Location: 15  
 Passages: 6,14,20  
 Moves: 10  
 Arrows left: 5

Figure 3: Game state after player is eaten by the wumpus

Figure 2: JS-Eden representation of the wumpus model

which, in the game implementation, is used as the observable referring to the room. I then labelled the map by using the room observables as points for the numbers. The whole of the map drawing is inter-dependent and any additional information that needs to go on it as the game progresses (such as the detection of bats and pits), is dependent on the room observables. The Wumpus’s location, as well as the locations of bats and pits, are also dependent on the room observables, and they are treated as observables themselves as they are part of the environment and they influence the course of the game.

## 6 Further work and other implementations of the game

It would be interesting to model the player’s approach to a ”Hunt The Wumpus” game that does not have the grid displayed, but rather the user is only aware of the paths accessible from their current location. In this case, the player would have to calculate what the other nearby rooms are, in an attempt to determine

where the positions of the game observables and for example decide if an obstacle found is the same one as one found earlier when traversing through the game grid in a different direction.

Another possible option would be to model the game when using a different shaped grid. It would be interesting to explore the strategies mentioned previously to find out how the shape of the game map influences the player’s decisions.

## Acknowledgements

I wish to acknowledge the work of Cole as the premise for adapting the existing eden model into the JS-Eden environment.

## References

- [1] G. Cole, Hunt the Wumpus: an Empirical Approach, Undergraduate fourth year module, 2005.
- [2] T. Monks, A Definitive System For The Browser, 2011

- [3] G. Yob, Hunt The Wumpus: The Genesis of Wumpus, 1976,
- [4] M. Beynon, (date unknown) Empirical Modelling for Educational Technology. University of Warwick, Coventry.
- [5] M. Beynon, (1999) Empirical Modelling and the foundations of Artificial Intelligence.