CSCI 446 Artificial Intelligence Project 2 Design Report

Roy Smart   Nevin Leh   Brian Marsh

October 7, 2016

1 Introduction

Logic and the Wumpus World is an artificial intelligence problem first proposed by Michael Genesereth, and described in detail by his student Stuart Russel[1]. The problem involves navigating an environment known as the Wumpus World using logic to avoid dangers as the agent attempts to reach a goal. The environment is a square grid of tiles that can be empty or contain gold for the goal state, a pit that the agent will fall into, or a monster known as a Wumpus. As the agent navigates the environment, a score is calculated from the various actions that the agent makes. The objective of the game is thus to maximize the score, which can be achieved by using logic to find the most efficient route.

2 Problem Statement

To solve the Wumpus World problem, we will develop a problem generator that creates the Wumpus worlds in which the agent will navigate. We will then implement a logic system that uses unification and resolution on first-order rules that allow the agent to navigate. The reasoning system will use the following information: the stench when near a Wumpus, the breeze when near a pit, the scream of a Wumpus as it is killed, and the wall when ran into. Additionally, a reactive agent will be created that makes decisions based upon random decisions regarding which cells are safe and which are not. We will test the logic system on Wumpus worlds of sizes 5x5, 10x10,âĂę, 25x25. Our performance metrics are as follows: number of times the gold is found, number of Wumpus killed, number of times the explorer falls into a pit, number of times the Wumpus kills the explorer, and number of cells explored.

3 Software Design

Our design will incorporate three main classes: `World`, `EnvironmentEngine`, and `Agent`. These classes have all of the important attributes and functions needed to implement this game. The `World` class is responsible for creating and maintaining worlds. The `EnvironmentEngine` is responsible for checking the move an agent selects and then telling the agent the result of the move. Finally, the `agent` is an abstract class that uses information provided by the `EnvironmentEngine` to decide what the next move is. The three implemented versions of `Agent` are `HumanAgent`, `ReactiveAgent` , and `ReasoningAgent`.

   The design of our software is centered around the environment engine. This class servers to link all components of the program and drives the main processes involved with this project.

3.1 Wumpus World Generation

We will implement a class

3.2 ENVIRONMENT ENGINE

3.3 REASONING AGENT

3.3.1 INFERENCE ENGINE

The core of this project is the inference engine. This is the part that will take our knowledge base and, as more information is added, infer more information to add to the knowledge base.

3.4 KNOWLEDGE BASE

3.5 PATHFINDING

3.6 REACTIVE AGENT

4 EXPERIMENT DESIGN

To analyze the performance of our agents, we are asked to vary various parameters of the wumpus World. These parameters include: the size of the board, and the number of wumpi, pits, and barriers.

REFERENCES

[1] Stuart Russel and Peter Norvig. *Artificial Intelligence: A Modern Approach.* Pearson Education, 3rd edition, 2010.