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,, 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

The software will center around an object denoted the `Environment_engine`. This class will be responsible for

3.1 Wumpus World Generation

The wumpus world will be represented

3.2 Environment Engine

3.3 Reasoning Agent

3.3.1 Inference Engine

Our inference engine will use the function PL-Resolution(KB, α) described by Russel and Norvig in Figure 7.12 [1] to perform *resolution*. Resolution allows the inference engine to use a knowledge base of facts

to ascertain new details about the wumpus world. The function PL-Resolution accepts a knowledge base, KB and α, the query to be checked. Of course, PL-Resolution is based on propositional logic clauses, but according to Section 9.5.2 of Russel and Norvig, we can adapt it to operate using first order logic clauses if we we modify the function PL-Resolve(C_i, C_j) to find two variables complementary if one *unifies* with the negation of the other.

To check if two variables can be unified, we will use the function Unify(x, y, θ) provided in Figure 9.1 of Russel and Norvig. In the interest of efficiency, we will also implement a *predicate indexing*, described in Section 9.2.3 of Russel and Norvig[1]. The predicate index provides Unify() an efficient way to only attempt unification on sentences that are possible to be unified. We will accomplish this by constructing lists of all facts that use a certain predicate, e.g. all facts that contain the predicate $Adjacent(r, s)$ will be placed into a list. Then, if Unify() is asked to unify a sentence containing $Adjacent(u, v)$, it will only have to search the list described in the previous sentence instead of the entire knowledge base.

3.3.2　Knowledge Base

3.3.3　Pathfinding

3.4　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.