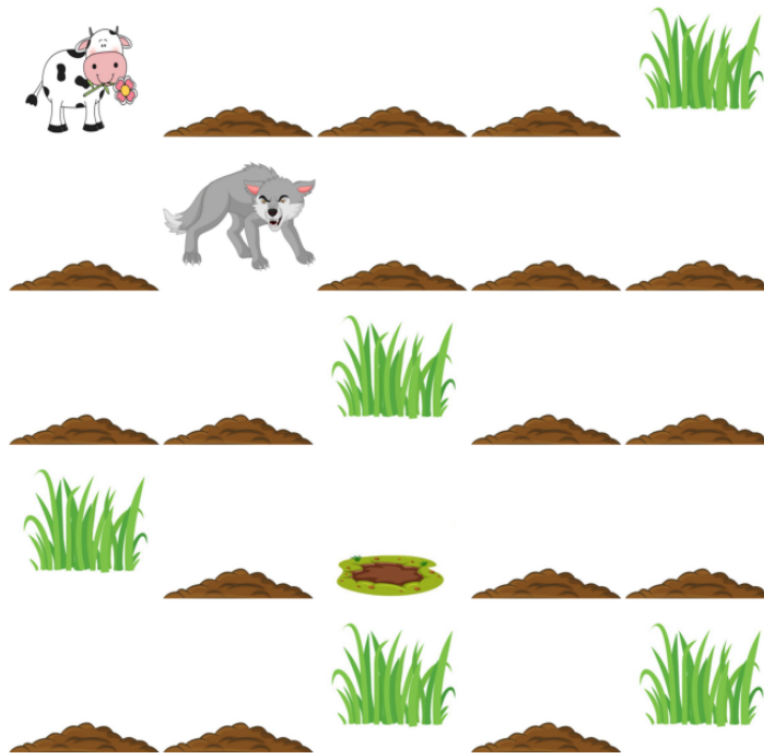**Ateneo De Manila University**
School of Science and Engineering

Title:

The Cow World

Meljohn Ugaddan
CSCI 211
Dr. Andrei Coronel and Sir Francis Bautista
August 8, 2021

A. Introduction

The Cow world is a world where there is a knowledge-based agent in a 5x5 grid environment making it a total of 25 blocks in the whole area that are connected to each other. The goal of our agent is that it should be able to satiate itself of 3 grasses as it explores the environment. As the agent traverses the environment, the agent must prevent going to an area where there is a predator and quicksand that will make our agent stuck there forever. The good thing is that the agent is able to perceive if there is a predator and quicksand near its area. Therefore, knowledge-based agents can infer if an area is safe to traverse.

B. PEAS Description
   a. Performance Measure
      i. Success : The game ends if the Cow agent was able to eat 3 grass
      ii. Fail : Was stuck in Quicksand or Eaten by wolf
   b. Environment
      i. A 5*5 grid of rooms.
      ii. The agent initially in room square [0, 0] or can be defined by the user
      iii. Location of Quicksand and Wolf will be static
   c. Actuators
      i. Down Turn
      ii. Up Turn
      iii. Right Turn
      iv. Left Turn
      v. Eat
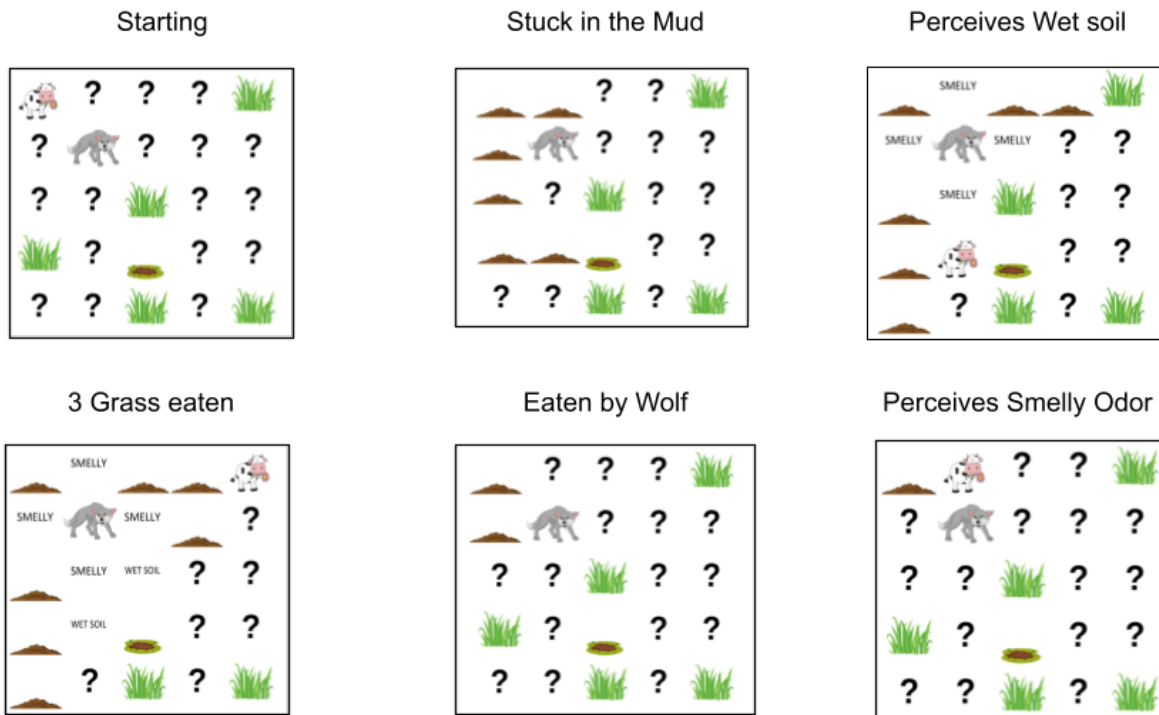   d. Sensors
      i. The cow agent will perceive the **smelly** in a surrounding in the area where wolf is near/adjacent
      ii. The agent will perceive **wetsoil** if the block is directly adjacent to the Quicksand.
      iii. The agent will perceive the **grass** if there is grass in the block
      iv. These sensor will be stored as a knowledge to use for inference to navigate safely in the world
   e. Properties
      i. Partially observable: The Cow Agent can only perceived the block area that where
      ii. Deterministic: It is deterministic, as the result and outcome of the world are already known. There are 3 outcomes, eaten by a wolf, stuck in mud and a cow agent is able to eat 3 grasses.
      iii. Sequential
      iv. Static: Quicksand and the Wolf position is fix
      v. Discrete: The environment is discrete since it is a 5x5 grid.
      vi. One agent: The Cow Agent

C. State Space

Starting

Stuck in the Mud

Perceives Wet soil

3 Grass eaten

Eaten by Wolf

Perceives Smelly Odor

Due to many combinations of the grid block, which can therefore result in many state spaces, I eliminated the redundancy. There are 6 important state spaces in the environment. The first one is the starting state, in this state there is no grass, smelly odor, and just soil. Next is when the cow agent perceives a wet soil. In this state the cow uses its knowledge from the environment/world to infer if its next destination is safe or not. This also applies to when an agent perceives a smelly odor. This knowledge will be used by the cow agent in order to prevent itself from being eaten by the wolf or will be stuck in the quicksand. The goal state is when the cow agent was able to eat three grasses.

D. State Transition Diagram
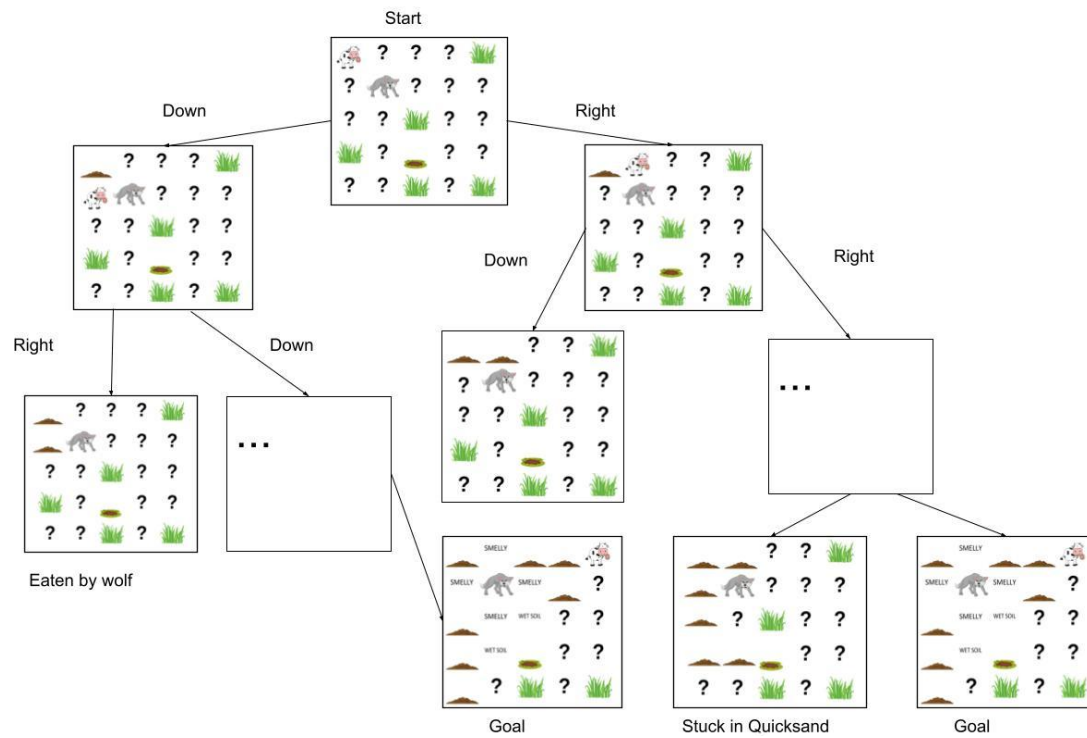
    Due to many combinations of the state-transition in our environment, I omitted some combinations. From the start picture, the agent starts at point (0,0) of the environment. It can be noted that the starting point can be also defined by the user. As it traverses to the right, it will go to the right. If the agent traverses downward, it will go in that direction. After all possible combinations of directions without stepping in quicksand or going to the wolf. It will achieve its goal where it was able to eat three grasses. If the agent goes to the area where there is a wolf, the agent will be eaten. The example combination in the diagram is from the start, then traverse to left, then right.

E. Percept-Action Mapping Table

Action	Percept/Percept Sequence
Cow Eats	Perceives a Grass
Cow use previous knowledge of the world then infer to avoid quicksand	Perceives a Wet Soil
Cow use previous knowledge of the world then infer to avoid getting eaten by wolf	Perceives a Smelly Odor

In the environment, if the cow perceives that there is grass in the area. It will eat the grass. If the agent perceives a smelly odor in an area it uses its previous knowledge to prove that the next block that it will traverse is safe. If not or the knowledge is insufficient it will traverse to different blocks. Same instance when the agent perceives that if it lands on wet soil, it must use its knowledge from the world to check if the next destination block is safe. The last is when the cow eats the third grass then it satiates its hunger.
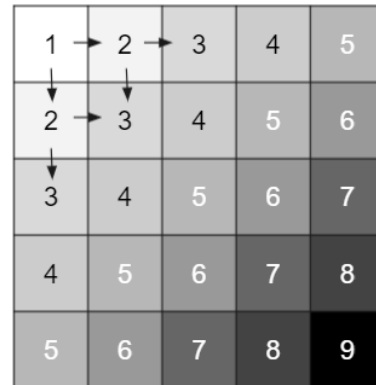
F. Agent's interaction with the environment

The first mode is the Depth-First Search. The example traversal is shown in the image below. Most of the implementation of the Depth-First Search in the grid is from the starting point, then traverses to the upper most part of the grid, then goes to left, then down, and lastly traverses to left.

1 →	2 →	3 →	4 →	5
22	21	14	13	6
23	20	15	12	7
24	19	16	11	8
25	18	17	10	9

21	20	3 →	4 →	5
22	19	2	13	6
23	18	1	12	7
24	17	14	11	8
25	16	15	10	9

The other traversing mode is in the image below. Most of the implementation of the Breadth-First Search in grid is from the starting point, then traverses to the adjacent grid.

<table>
<tr><td>5</td><td>4</td><td>3</td><td>4</td><td>5</td></tr>
<tr><td>4</td><td>3 ← 2 →</td><td></td><td>3</td><td>4</td></tr>
<tr><td>3 ←</td><td>2 ← 1 →</td><td></td><td>2 →</td><td>3</td></tr>
<tr><td>4</td><td>3 ← 2 →</td><td></td><td>3</td><td>4</td></tr>
<tr><td>5</td><td>4</td><td>3</td><td>4</td><td>5</td></tr>
</table>

<table>
<tr><td>1 →</td><td>2 →</td><td>3</td><td>4</td><td>5</td></tr>
<tr><td>2 →</td><td>3</td><td>4</td><td>5</td><td>6</td></tr>
<tr><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td></tr>
<tr><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td></tr>
<tr><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td></tr>
</table>
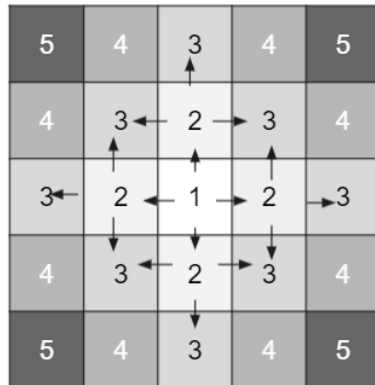
For the inference whenever the agent uses the knowledge to traverse the environment safely. There are two propositional rules that are used by the agent to prove that the next block to be traversed will be safe. The first rule is that if a block *(i,j)* has a smelly odor, then there must be a wolf on its adjacent block. This rule also applies to quicksand. The second rule states that there is a wolf block *(i,j)* if there is smelly on a diagonally aligned adjacent block *(i,j-1) and (1-j,j)* provided that (i-1,j-1) block is visited and safe. This rule also applies to the other 3 combinations and quicksand.

1. $Smelly_{i,j} \longrightarrow Wolf_{i-1,j}$ OR $Wolf_{i,j-1}$ OR $Wolf_{i+1,j}$ OR $Wolf_{i,j+1}$
2. $Wolf_{i,j} \longrightarrow Visited_{i-1,j-1}$ AND $Smelly_{i-1,j}$ AND $Smelly_{i,j-1}$

<table>
<tr><td>(i-1 , j-1)</td><td>(i-1 , j)</td><td>(i-1 , j+1)</td></tr>
<tr><td>(i , j-1)</td><td>(i , j)</td><td>(i,j+1)</td></tr>
<tr><td>(i+1,j-1)</td><td>(i+1, j)</td><td>(i+1,j+1)</td></tr>
</table>