CS440: AI Project 3 Wumpus Fog of War World Lab Report

Christopher Yong (cy287), Fares Elkhouli (fae15),
Shlomo Benyaminov (ssb165),
Dennis Galinsky (Dbg65)

December 2, 2020

General Information

How to play?

1) First set the dimension of the board

2) Move your piece

3) Press "Next Turn" button

4) Repeat Steps 2-3 till Game is over

Note To see the probabilities, right click on the cell.
To see the AI fog of war map, press AI fog
To see Player fog of war map, press Player fog
To see full state of the map, press No fog

Dependencies

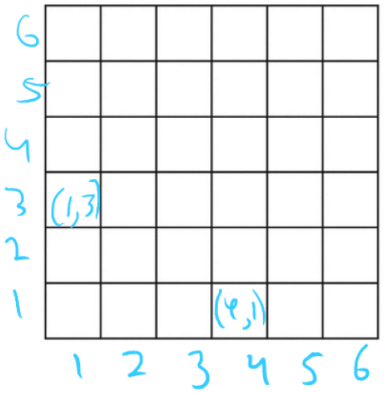
- JavaFX

Question 1

Formulate the new wumpus game in terms of Boolean variables and define the relationship between these variable. Hint: For each cell you should have a variable for each observation and a variable for each item that the cell can contain (i.e. units, pits). (10 points)

Answer:

Board indices: (X, Y)



① Start game

AI pieces: $W_{1,1}$ $H_{2,1}$ $M_{3,1}$ $W_{4,1}$ $H_{5,1}$ $M_{6,1}$

Player pieces: $W_{1,6}$ $H_{2,6}$ $M_{3,6}$ $W_{4,6}$ $H_{5,6}$ $M_{6,6}$

Pits: $P(Pit_{x,y}) = \frac{1}{6}$ for $2 \leq X \leq 5$ ∩ $1 \leq Y \leq 6$

Breeze - if adjacent neighbor is a pit

$$\text{Breeze}_{x,y} \leftrightarrow \left(\begin{array}{l} P_{x-1,y+1} \vee P_{x,y+1} \vee P_{x+1,y+1} \\ \vee P_{x+1,y} \vee P_{x+1,y-1} \vee P_{x,y-1} \\ \vee P_{x-1,y-1} \vee P_{x-1,y} \end{array} \right)$$

where
$1 \leq x \leq 6$
$1 \leq y \leq 6$

Stench - if adjacent neighbor is an enemy wumpus

$$\text{Stench}_{x,y} \leftrightarrow \left(\begin{array}{l} W_{x-1,y+1} \vee W_{x,y+1} \vee W_{x+1,y+1} \\ \vee W_{x+1,y} \vee W_{x+1,y-1} \vee W_{x,y-1} \\ \vee W_{x-1,y-1} \vee W_{x-1,y} \end{array} \right)$$

where
$1 \leq x \leq 6$
$1 \leq y \leq 6$
and
$W_{x',y'}$ is an enemy wumpus

Noise - if adjacent neighbor is an enemy Hero

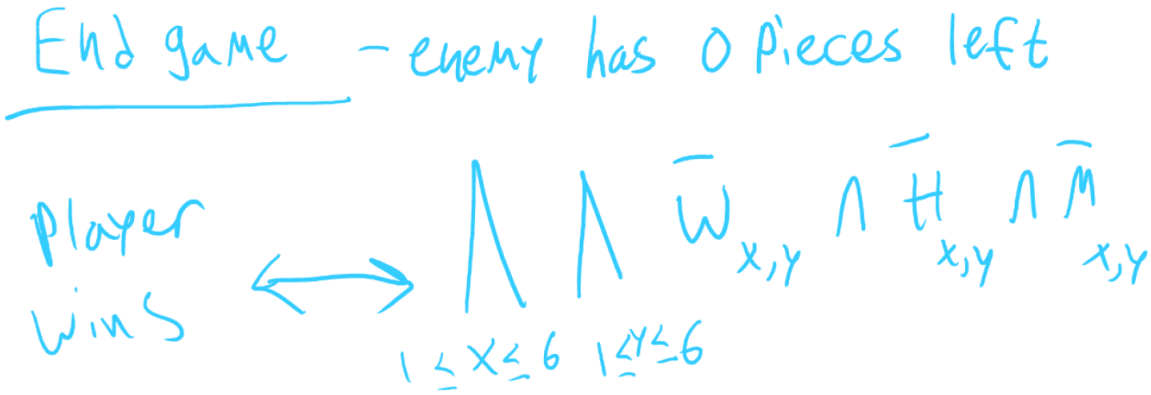
$$\text{Noise}_{x,y} \leftrightarrow \begin{pmatrix} H_{x-1,y+1} & \vee H_{x,y+1} & \vee H_{x+1,y+1} \\ \vee H_{x+1,y} & \vee H_{x+1,y-1} & \vee H_{x,y-1} \\ \vee H_{x-1,y-1} & \vee H_{x-1,y} & \end{pmatrix}$$

where $1 \leq x \leq 6$, $1 \leq y \leq 6$, and $H_{x',y'}$ is an enemy hero

Heat - if adjacent neighbor is an enemy Mage

$$\text{Heat}_{x,y} \leftrightarrow \begin{pmatrix} M_{x-1,y+1} & \vee M_{x,y+1} & \vee M_{x+1,y+1} \\ \vee M_{x+1,y} & \vee M_{x+1,y-1} & \vee M_{x,y-1} \\ \vee M_{x-1,y-1} & \vee M_{x-1,y} & \end{pmatrix}$$

where $1 \leq x \leq 6$, $1 \leq y \leq 6$, and $M_{x',y'}$ is an enemy Mage



Question 2

Update your game interface to display the observations and to include the fog of war option that can be toggled on and off to display and hide the agent's pieces. (10 points)

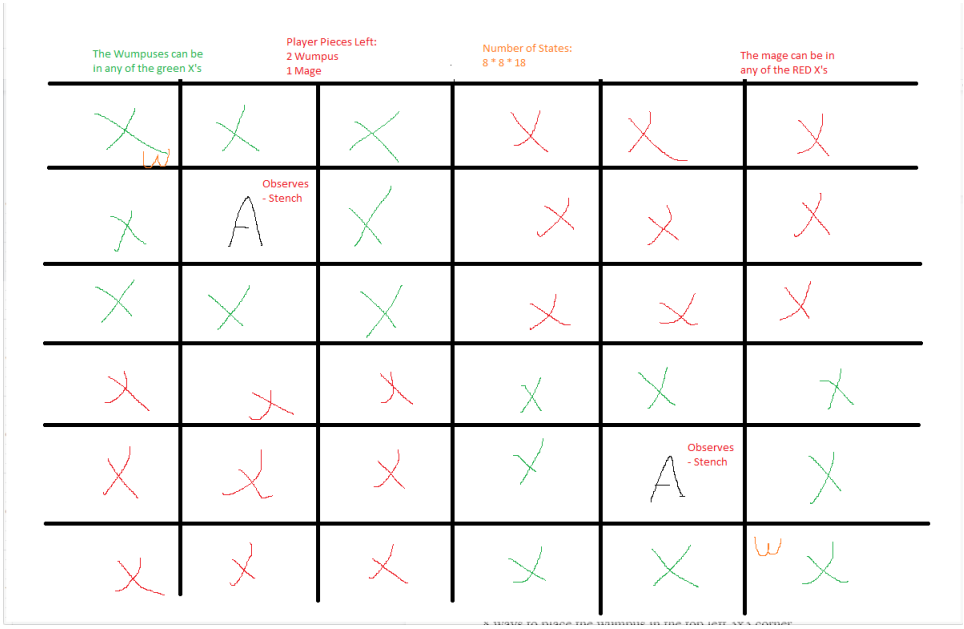
Answer:
Implemented in code.

Question 3

Implement the method for computing probability that you described in section 3. Modify your interface to display the probabilities P(Px,y), P(Px,y), P(Hx,y) and P(Mx,y) for each cell. (20 points)

Answer:
Implemented in code.

Overview of the probability calculations:



The number of states that will give us this observation is:
8 ways to place the wumpus in the top left 3x3 corner
8 ways to place the wumpus in the bottom right 3x3 corner
18 ways to place the mage in the other cells.
Therefore there are 8 * 8 * 18 numbers of states that will give us this observation.

Question 4

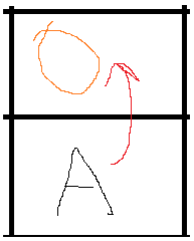
Propose a policy for selecting a good move given a probability distribution. Implement an agent that uses this policy. (30 points)

Answer:

Policy:

1) Never check a pit that has a probability greater than zero because with observations, we can predict cells that do not have any probability of being a pit.

2) Moving a piece into cell (x,y)

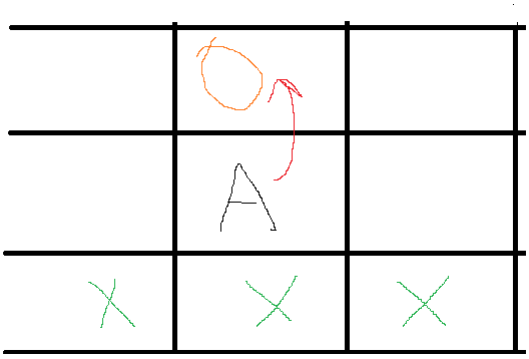


Case 1) Cell (x,y) has a piece that is killable therefore we give this move a positive value

Case 2) Cell (x,y) has a piece that will kill our piece therefore we give this move a negative value

Case 3) Cell (x,y) has a piece that will kill both pieces and in this case, we give this move a positive value

3) Moving a piece into cell (x,y) moves us away from cells



Case 1) The cells we are escaping have a chance of a piece that can kill our piece therefore moving away from those pieces gives us a positive value

Case 2) The cells we are escaping have a chance of a piece we can kill therefore moving away from the pieces gives us a negative value

Case 3) The cells we are escaping are the same piece as ours therefore we treat this as a neutral exchange where there is no loss or benefit from moving away.

Question 5

In Section 2 we assume that the opponent will randomly select a piece to move and then move that piece in a random direction. This is not a realistic approximation of the moves the opponent will select.Propose a better alternative for

approximating the opponents moves and discuss how you would compute the effect of the opponent's move on the probability distribution given this approximation method. (10 points)

Answer:

Instead of assuming a player makes a random move which is unrealistic, we will assume that the player always tries to minimize the distance from its piece to a killable AI enemy piece. For example, if the player has a Wumpus, the player will try to get closer to the AI's Mage so that it could kill it.

The diagram below demonstrates this pattern. On this grid, the AI has 3 pieces left, a Wumpus, Mage, and Hero. The AI also knows the probability distribution of the current state and board. What we will do here is come up with weights for P(W), P(H), and P(M) for every cell, then multiply P(W), P(H), and P(M) with their respective weights to get an updated probability distribution that favors minimizing a player's piece to a killable AI piece.

The calculation for the P(W) weight is shown below as a sample. Note that similar calculations are calculated for P(H) and P(M).

$$\text{weighted probability for P}(W_{x,y}) = 1 - \frac{MinDistancefromx,ytoAIMage}{(sizeofgrid)*\sqrt{2}}$$
$$\text{weighted probability for P}(H_{x,y}) = 1 - \frac{MinDistancefromx,ytoAIWumpus}{(sizeofgrid)*\sqrt{2}}$$
$$\text{weighted probability for P}(M_{x,y}) = 1 - \frac{MinDistancefromx,ytoAIHero}{(sizeofgrid)*\sqrt{2}}$$

Note that (size of grid) * sqrt(2) is the maximum distance between any 2 pieces in the grid. This is used to normalize the value of the weight to be in between 0 and 1, to ensure probabilities make sense.

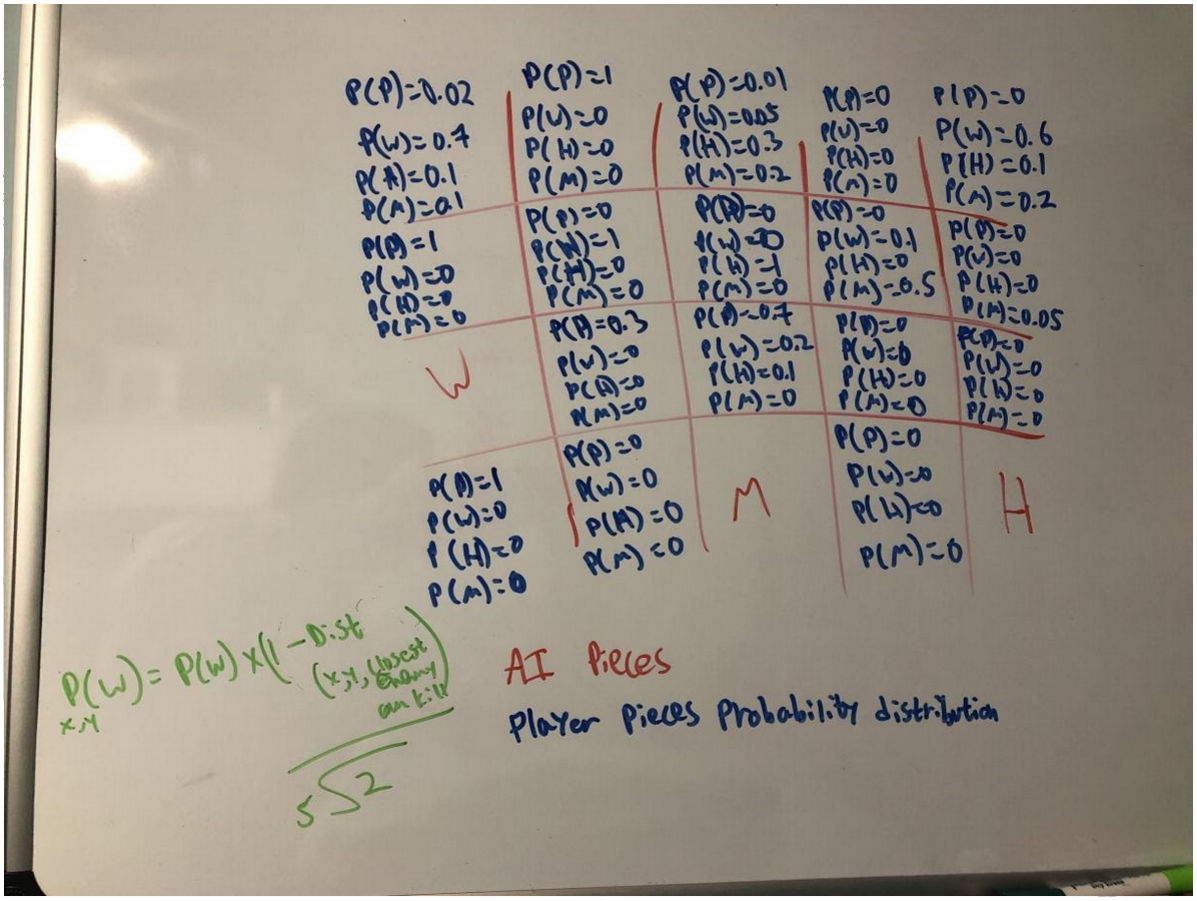
$$\text{P}(W_{x,y}) = (\text{weight for P}(W_{x,y})) * \text{P}(W_{x,y})$$
$$\text{P}(H_{x,y}) = (\text{weight for P}(H_{x,y})) * \text{P}(H_{x,y})$$
$$\text{P}(M_{x,y}) = (\text{weight for P}(M_{x,y})) * \text{P}(M_{x,y})$$

This calculation will ensure that cells closer to an AI piece the player can kill will have a greater probability of occuring, hence a better probability distribution than assuming a player moves randomly.

However, after this process, we have to re-normalize the probability for a Wumpus / Mage / Hero for all other cells since we are using weight-probabilities. The equation below will perform the normalization processes.

$$\text{P'}(W_{x,y}) = \frac{P(W_{x,y})}{\sum_{allx'y'ingame} P(W_{x',y'})}$$
$$\text{P'}(H_{x,y}) = \frac{P(H_{x,y})}{\sum_{allx'y'ingame} P(H_{x',y'})}$$

$$\mathrm{P'}(\mathrm{M}_{x,y}) = \frac{P(M_{x,y})}{\sum_{allx'y'ingame} P(M_{x',y'})}$$



Question 6

Implement the transition probability method that your described in question 5. (20 points)

Answer:
Implemented in code.