INFO8006: Project 2 - Report

Maxime Goffart - s180521
Olivier Joris - s182113

October 19, 2020

1 Problem statement

a.
- (Initial state) A game state is given by the position of Pacman, the position of the ghost, and the positions of the remaining food dots in the maze.
  The initial state is given by the layout of the maze, the initial position of Pacman, the initial position of the ghost, and the initial positions of all the food dots in the maze.

- (Player function) The player function is a function that takes a state as input and returns a value corresponding to the player who has to play. It respects this principle : an agent takes an action then the other takes one and so on. It is a turn taking game.

- (Actions) Pacman and the ghost can go north, south, east, or west if they don't go through a wall. Both of them can, also, stay on the same cell.
  If Pacman arrives on a cell with a food, Pacman eats the food.
  If the ghost arrives on the cell where Pacman is, the ghost kills Pacman.

- (Terminal test) True if Pacman has eaten all the foods without being killed or Pacman was killed by the ghost.

- (Transition model) The movement of Pacman or the ghost to another cell will modified their position in the maze (if the movement is legal). If Pacman eats a food, the food is removed from the maze. If the ghost kills Pacman, Pacman is removed from the maze and can no longer play.

- (Utility)
$$utility = \begin{cases} \text{Game score} & \text{Pacman wins (ghost loses)} \\ -\text{Game score} & \text{Pacman loses (ghost wins)} \end{cases} \tag{1}$$

b. Pacman will be the max agent and the ghost the min agent. Pacman wants to maximize the utility function (which implies maximizing the game score) by eating all the food dots in a minimum time steps and avoiding the ghost.
   The ghost, on his side, wants to kill Pacman as fast as possible to minimize the utility function (which implies minimizing the game score). This description make the game of Pacman a zero-sum game because the sum of the utility functions in the different terminal states is equal to 0.

2 Implementation

a. Minimax is complete if the game tree is finite. For the game Pacman, in general, the game tree is finite because either Pacman will eat all the food dots or the ghost will kill Pacman. However, if the ghost keeps chasing Pacman without being able to kill it and Pacman is not able to eat all the food dots, the algorithm will never end.
   As for the A* graph-search algorithm, we could keep a set which will remember the game states already considered so we don't spend computing time to recompute them. This will prevent Pacman and the ghost to do the same movements indefinitely.

b. ***Leave empty.***

c. *Leave empty.*

d.

3   Experiment

a.

b.

c.