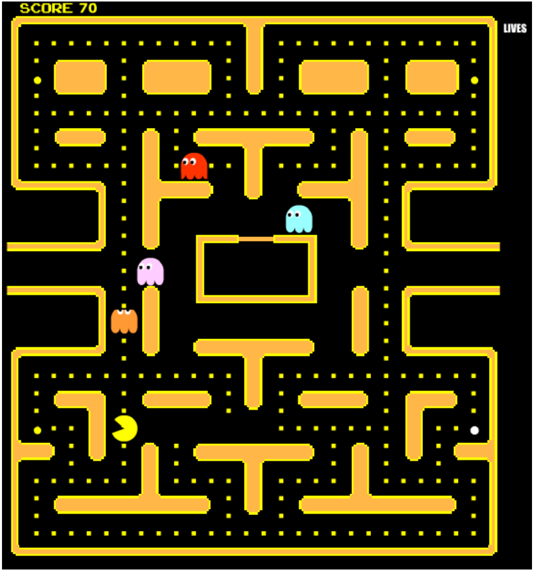
PACMAN and reinforcement learning



PACMAN and RL - Goals

- Model the behaviour of game characters: ghost
- The usual way: Program by hand the decisions of the character
- Drawbacks:
  - Difficult to program (lots of possible states)
  - Difficult to adapt the behaviour to the human player
- Solution: Learn how to act by training the character
- Reinforcement learning allows to train a decision mechanism from sequences of actions obtained from character-player interaction

PACMAN and RL - what do we need?

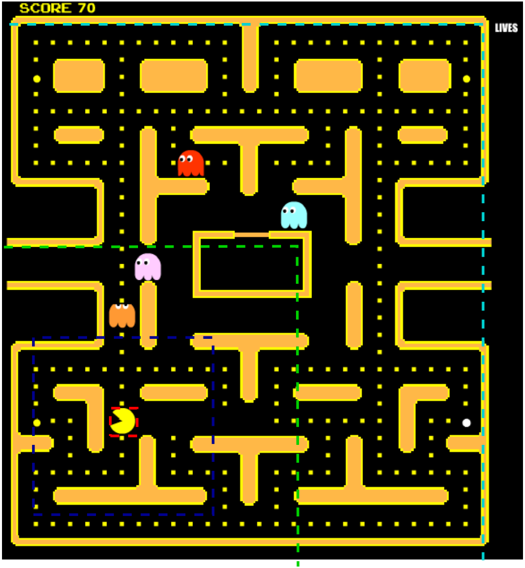
1. To solve a problem using reinforce learning we need to formalize it using a decision markow process: states and actions
    - States: Positions of the ghost + state of PACMAN (vulnerable, invulnerable)
    - Actions: Possibles movements of the ghost
2. We have to decide what states will generate a reward:
    - Ghost catches PACMAN vulnerable → positive reward
    - PACMAN invulnerable catches ghosr→ negative reward
3. We have to decide what to use for training: sequences of actions of the ghost

PACMAN and RL - Simple Model (I)

- We can model the problem at different levels of granularity
- We can consider the model from RL as a high level decision mechanism that gives an indication of the action to perform and another decision mechanism gives the primitive actions to achieve it
- Modelo simple:
  - State: $\{> 10, > 5, \leq 5, \text{PACMAN}\} \times \{\text{Vulnerable, Invulnerable}\}$
  - Acciones:

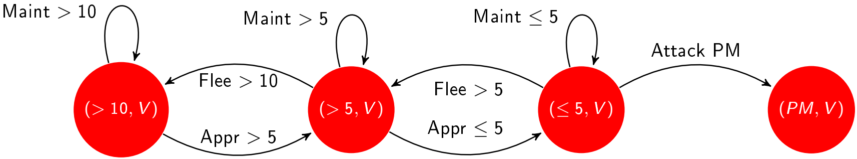
$$\mathit{Actions} \left\{ \begin{array}{lll} \text{Approach} > 5 & \text{Approach} \leq 5 & \\ \text{Atack PACMAN} & & \\ \text{Flee} > 10 & \text{Flee} > 5 & \\ \text{Maintain} > 10 & \text{Maintain} > 5 & \text{Maintain} \leq 5 \end{array} \right.$$

PACMAN and RL - Simple Model (II)



PACMAN and RL - Simple Model (III)

- Whe have two sets of states
- For the case of vulnerable PACMAN:



PACMAN and RL - Simple Model (IV)

- The training sequences will be sequences of ghost movements
- Problem: Not al sequences end in a goal state
- We can add prositive and negative rewards in the rest of the states to be able to use all training sequences
  - Approach vulnerable PACMAN → positive
  - Approach invulnerable PACMAN → negative

PACMAN and RL - Detailed Model (I)

- We can model the problem so the decisions are primitive actions
- Detailed model:
  - States: Product set of the coordinates of PACMAN and the ghost and the state of PACMAN
  - Actions: up, down, left, right

PACMAN and RL - Detailed Model (II)

- We will have a model for the states when PACMAN is vulnerable and other when is invulnerable
- The goal state will be those where the coordinates of PACMAN and the ghost are the same (distance = 0)
- The reward can be a function of the distance
- Drawbacks:
    - The model is large (but can be indexed)
    - The number of training sequences to obtain a good model could be very large
- Advantages:
    - When the model converges the behaviour will be to approach or flee PACMAN using the shortest path