Assignment 6

Maxime CHAMBREUIL
McGill ID: 260067572
maxime.chambreuil@mail.mcgill.ca

Contents

1 Expected Utility

1.1 Decision graph

Description of the variables, actions and rewards used:

- Neuro Pb : binary variable, stands for the existence of a neurotransmitter problem
- Minor Pb : binary variable, stands for the existence of a minor problem
- Softw Pb : binary variable, stands for the existence of a software problem
- Use Analyzer : binary variable, stands for the decision to use or not the analyzer
- Damage : negative reward due to damages that costs 70$
- Result : binary variable, if Data has won or lost the contest
- Know Pblem : binary variable, stands for the knowledge of the problem or not
- Repair : negative reward due to repairing that costs 20$

- Enter or not : binary variable, stands for the decision to enter or not the contest
- Money : positive (100$) or nul (0$) reward due to the result of the contest
- Entr fee : negative reward due to the fee Data has to pay to enter the contest

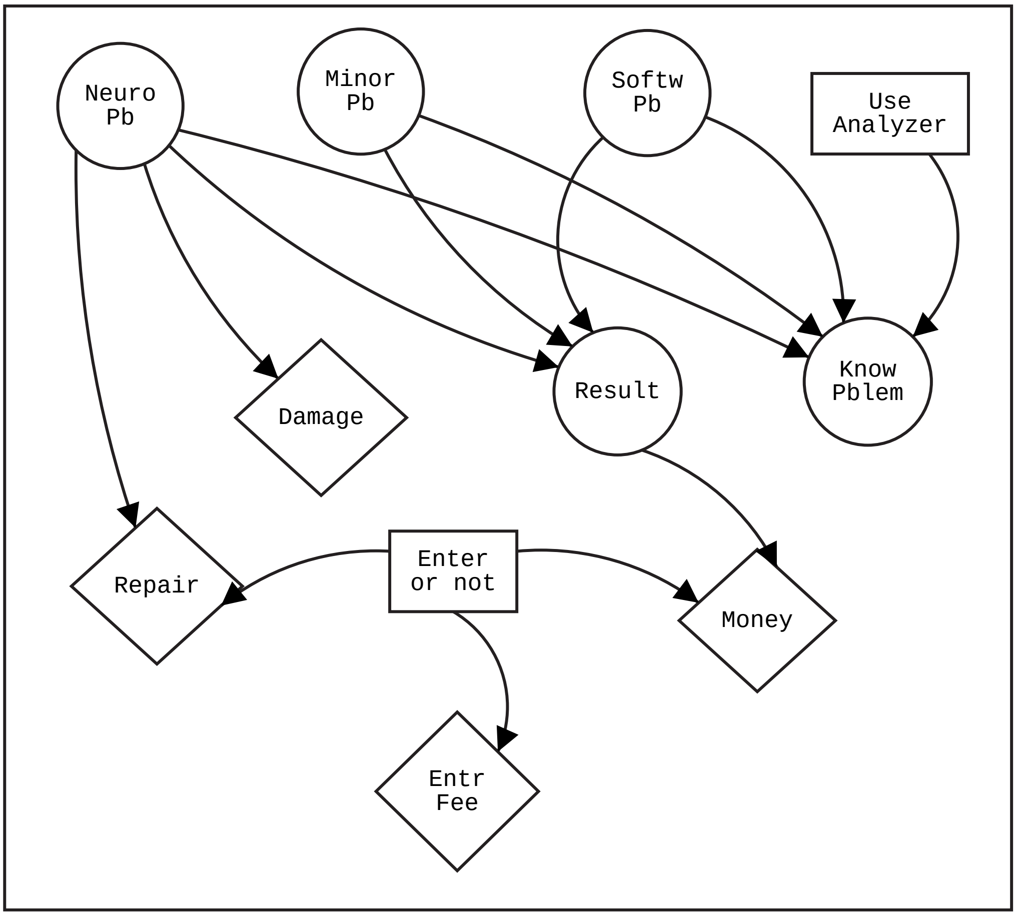

Figure 1: Decision graph

1.2 Entrance fee

The expected value if Data enters the contest is:

$$E = 0.3 \times (-70) + 0.3 \times 0.5 \times 100 + 0.3 \times 0.5 \times 0 + 0.4 \times 100 \times 0.9 + 0.3 \times 0 = 30\$$$

If Data does not enter the contest and has a neurotransmitter problem, he has to pay :

$$0.3 \times 20 = 6\$$$

So Data should enter the contest if the entrance fee is not more than 36$: If the entrance fee is 35$, Data enters the contest and can expect to win 30$, so he will lose only 5$ instead of 6 if it does not enter.

1.3 Analyzer use

The analyzer is useless as its result will not influence our decision to enter the contest or not.

2 Markov Decision Problem

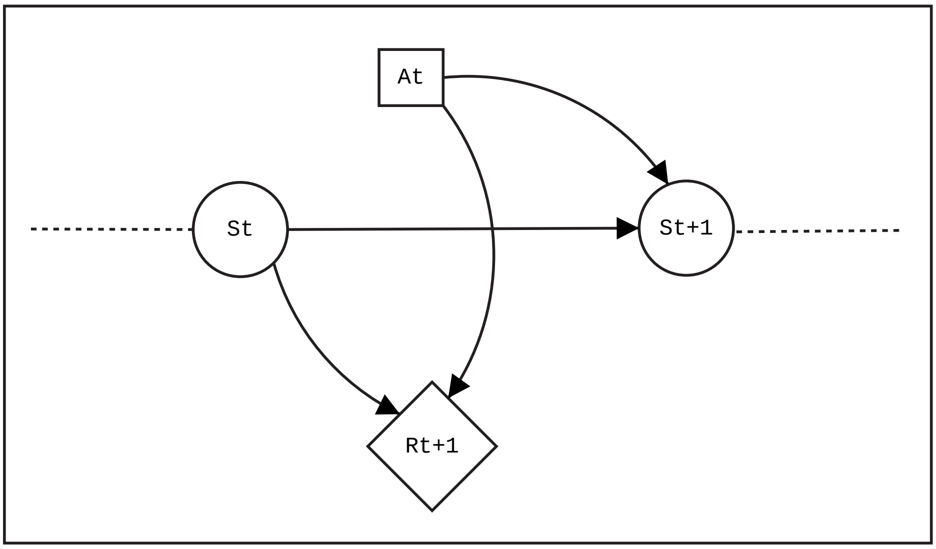

Figure 2: MDP schema

To formulate this problem as a Markov Desion Process, we need to define:

- State : Number of cars sold k and number of cars not sold i
- Action : Buy a certain number of cars a
- Reward : $R = c \times k - u \times i - a \times d$
- Transition probabilities : probability to go from a couple (k,i) to any other couple (k,i), with k and i positive integers

3 Optimal policies

With an horizon greater or equals to 3, the best action when you are on the goal state is to stay.

The actions are: up, down, right, left and stay. The reward is one in the goal state and zero otherwise. The transition probability to go from one state to another is one over your number of neighbors plus one (you can stay). For example :

$$p(S_{t+1} = G/S_t = G) = \frac{1}{3+1} = \frac{1}{4}$$

$$p(S_{t+1} = G/S_t = 3) = \frac{1}{2+1} = \frac{1}{3}$$

4 Action Values

4.1 Belman equation

$$\begin{aligned} Q^\pi(S,a) &= E\{r_1 + \gamma r_2 + \ldots / S_0 = S, a_t = a, \text{ follow } \pi\} \\ &= R(S,a) + \gamma \sum_{S'} P^a_{SS'} V^\pi(S') \\ &= R(S,a) + \gamma \sum_{S'} P^a_{SS'} \sum_a \pi(S',a) Q^\pi(S',a) \end{aligned}$$

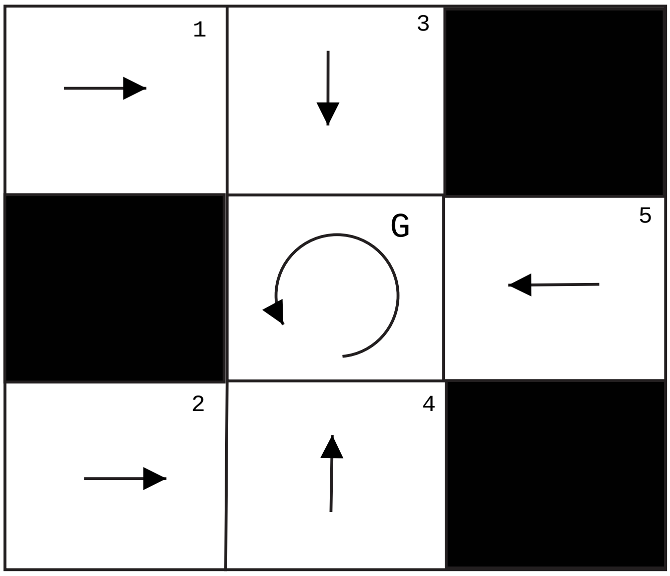

Figure 3: Environment

4.2 TD learning rule

$$\hat{Q}^{\pi}(S_t, a) \leftarrow \alpha \left[r_{t+1} + \gamma \hat{Q}^{\pi}(S_{t+1}, a) \right] + (1-\alpha)\hat{Q}^{\pi}(S_t, a)$$

5 Reinforcement learning

5.1 Advantage function in terms of action values

$$\begin{aligned} A^{\pi}(S,a) &= Q^{\pi}(S,a) - V^{\pi}(S) \\ &= Q^{\pi}(S,a) - \sum_a \pi(S,a) Q^{\pi}(S,a) \end{aligned}$$

5.2 2 actions a_1 and a_2

$$\begin{aligned} A^{\pi}(S,a_1) &= Q^{\pi}(S,a_1) - V^{\pi}(S) \\ A^{\pi}(S,a_2) &= Q^{\pi}(S,a_2) - V^{\pi}(S) \end{aligned}$$

By substracting on each side, we obtain:

$$A^{\pi}(S,a_1) - A^{\pi}(S,a_2) = Q^{\pi}(S,a_1) - Q^{\pi}(S,a_2) = 0 \Rightarrow A^{\pi}(S,a_1) = A^{\pi}(S,a_2)$$

As the 2 actions have the same probabilities, there is no advantage to choose one or the other.

5.3 Best action a^*

By definition of $V^{\pi}(S)$, we have:

$$A^{\pi}(S,a^*) = Q^{\pi}(S,a^*) - V^{\pi}(S) = 0$$

6 Exploration

6.1 Environment

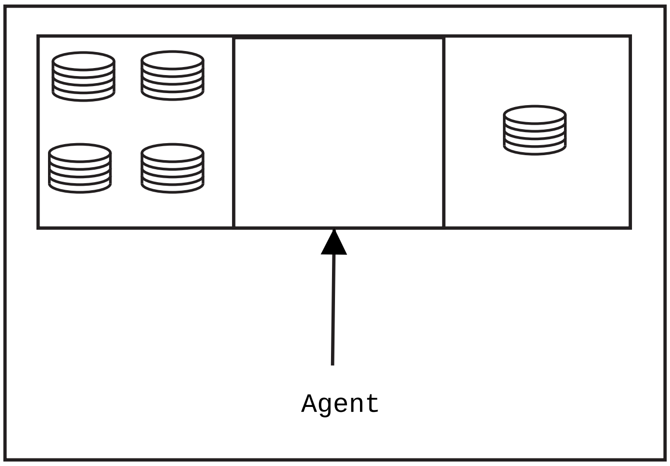

Figure 4: Environment

If the agent goes right at the beginning, it is going to try left faster with the greedy method than with ε-greedy: With ε-greedy, the policy will affect ε to the ”going left” action and would not go left as fast as the greedy action.

6.2 Best strategy

The optimistic initial value/greedy action would perform better as it finds the best action in a shorter time than ε-greedy.