EL2805 Reinforcement Learning

Computer Lab 1

The Great Escape

November 13, 2018

Department of Automatic Control
School of Electrical Engineering and Computer Science
KTH Royal Institute of Technology

Instructions (read carefully):

- Solve at least Problems 1 and 3. Solving Problem 2 correctly gives you 1 extra point at the exam (out of 50 points).
- Work in groups of 2 persons.
- Write a joint report where you answer the questions and include relevant figures.
  - **Include both persons' names and personal numbers in the report.**
  - Name the file as follows:

    `LASTNAME1-FIRSTNAME1-LASTNAME2-FIRSTNAME2-Lab1.pdf` for the report,

    where `FIRSTNAME1` is the firstname of Student 1 in the group (etc.).
  - Preferably, use the NIPS template for the report:
    `https://nips.cc/Conferences/2018/PaperInformation/StyleFiles`
  - Hand-written solutions will not be corrected.
- **Both** students in the group should upload the report as a .pdf-file to Canvas before November 30, 23:59. The deadline is strict.

Good luck!

Problem 1:
The Maze and the Random Minotaur

Consider the maze in Figure 1. You enter the maze in A and at the same time, the minotaur enters in B. The minotaur follows a random walk while staying within the limits of the maze. The minotaur's walk goes through walls (which obviously you cannot do). At each step, you observe the position of the minotaur, and decide on a one-step move (up, down, right or left) or not to move. If the minotaur catches you, he will eat you.[1] Your objective is to identify a strategy maximising the probability of exiting the maze (in B) before time T. *Note:* Neither you nor the minotaur can walk diagonally.

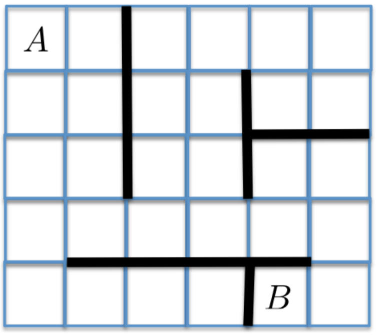

Figure 1: The minotaur's maze.

(a) Formulate the problem as an MDP.

(b) Solve the problem, and illustrate an optimal policy for $T = 15$.[2] Plot the maximal probability of exiting the maze as a function of T. Is there a difference if the minotaur is allowed to stand still? If so, why?

(c) Assume now that your life is geometrically distributed with mean 30. Modify the problem so as to derive a policy minimising the expected time to exit the maze. Motivate your new problem formulation (model). Estimate the probability of getting out alive using this policy by simulating 10 000 games.

[1] `https://en.wikipedia.org/wiki/Minotaur`

[2] *Hint:* To illustrate a policy, you could, for example; simulate a game and show the steps taken, plot the action in each player position for a fixed minotaur position, or something else. Be creative.

Problem 2:
Robbing Banks

At time 0, you are robbing Bank 1 (see Figure 2), and the police gets alerted and starts chasing you from the point PS (Police Station). You observe where the police is, and decide in each step either to move up, left, right, down or to stay where you are. Each time you are at a bank, and the police is not there, you collect a reward of 10 SEK. If the police catches you, you loose 50 SEK, and the game is reinitialised (you go back to Bank 1, and the police goes back to the PS).

The police always chases you, but moves randomly in your direction. More precisely, assume that the police and you are on the same line, and without loss of generality, that the police is on your right; then the police moves up, down and left with probability 1/3. Similarly, when the police and you are on the same column, and when your are above the police, then the police moves up, right and left with probability 1/3. When the police and you are not on the same line, nor on the same column, say the police is below you and on your right, then the police moves up and left with probability 1/2.

The rewards are discounted at rate $\lambda \in (0, 1)$, and your objective is to maximise your average discounted reward.

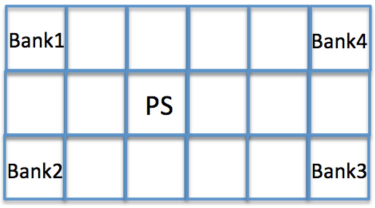

Figure 2: The city where you are robbing banks.

(a) Formulate the problem as an MDP.

(b) Solve the problem, and display the value function (evaluated at the initial state) as a function of λ. Illustrate an optimal policy for different values of λ – comment on the behaviour.

Problem 3:
Bank Robbing (Reloaded)

You are a bank robber trying to heist the bank of an unknown town. You enter the town at position A (see Figure 3), the police starts from the opposite corner, and the bank is at position B. For each round spent in the bank, you receive a reward of 1 SEK. The police walks randomly (that is, uniformly at random up, down, left or right) across the grid and whenever you are caught (you are in the same cell as the police) you lose 10 SEK.

You are new in town, and hence oblivious to the value of the rewards, the position of the bank, the starting point and the movement strategy of the police. Before you take an action (move up, down, left, right or stand still), you can observe both your position and that of the police. Your task is to develop an algorithm learning the policy that maximizes your total discounted reward for a discount factor $\lambda = 0.8$.

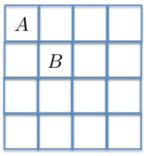

Figure 3: The unknown town.

(a) Solve the problem by implementing the Q-learning algorithm exploring actions uniformly at random. Create a plot of the value function over time (in particular, for the initial state), showing the convergence of the algorithm. **Note:** Expect the value function to converge after roughly 10 000 000 iterations (for step size $1/n(s,a)^{2/3}$, where $n(s,a)$ is the number of updates of $Q(s,a)$).

(b) Solve the problem by implementing the SARSA algorithm using ε-greedy exploration (initially $\varepsilon = 0.1$). Show the convergence for different values of ε.