Intelligent Autonomous Agents and Cognitive Robotics
Exercise Sheet 9

1. Sometimes MDPs are formulated with a reward function R(s, a) that depends on the action taken or with a reward function R(s, a, s') that also depends on the outcome state.

   a. Write the Bellman equations for these formulations.

   b. Show how an MDP with reward function R(s, a, s') can be transformed into a different MDP with reward function R(s, a), such that optimal policies in the new MDP correspond exactly to optimal policies in the original MDP.

   c. Now do the same to convert MDPs with R(s, a) into MDPs with R(s)

2. In this exercise we will consider two-player MDPs that correspond to zero-sum, turn taking games. Let the players be A and B, and let R(s) be the reward for player A in s. (The reward for B is always equal and opposite.)

   a. Let $U_A(s)$ be the utility of state s when it is A's turn to move in s, and let $U_B(s)$ be the utility of state s when it is B's turn to move in s. All rewards and utilities are calculated from A's point of view (just as in a minimax game tree). Write down Bellman equations (equations used for value iteration) defining $U_A(.s)$ and $U_B(s)$.

   b. Explain how to do two-player value iteration with these equations, and define a suitable stopping criterion.

   c. Consider the game described in the following figure. Draw the state space (rather than the game tree), showing the moves by A as solid lines and moves by B as dashed lines. Mark each state with R(s). You will End it helpful to arrange the states (s_A, s_B) on a two-dimensional grid, using s_A and s_B as "coordinates."

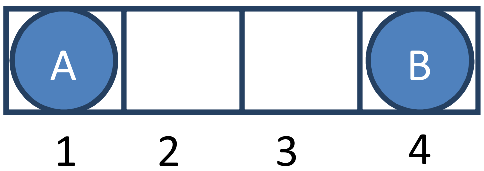

Figure 4.
The starting position of a simple game. Player A moves first. The two players take turns moving, and each player must move his token to an open adjacent space in either direction. If the opponent occupies an adjacent space, then a player may jump over the opponent to the next open space if any. (For example, if A is on 3 and B is on 2, then A may move back to 1.) The game ends when one player reaches the opposite end of the board. If player A reaches space 4 first, then the value of the game to A is +1; if player B reaches space 1 first, then the value of the game to A is -1.

d. Now apply two-player value iteration to solve this game, and derive the optimal policy.

3. Give the pseudo code for policy iteration. Explain how the major steps can be implemented.

4. Consider an undiscounted Markov Decision Process (MDP) having three states (1,2,3), with rewards -1, -2, 0 respectively. State 3 is a terminal state. In states 1 and 2 there are two possible actions: a and b. The transition model is as follows:

   - In state 1, action a moves the agent to state 2 with probability 0.8 and makes the agent stay put with probability 0.2
   - In state 2, action a moves the agent to state 1 with probability 0.8 and makes the agent stay put with probability 0.2
   - In either state 1 or state 2, action b move the agent to state 3 with probability 0.1 and makes the agent stay put with probability 0.9

   Answer the following questions:

   a. What can be determined qualitatively about the optimal policy in state 1 and state 2?
   b. Apply policy iteration, showing each step in full, to determine the optimal policy and the values of state 1 and state 2. Assume that the initial policy has action b in both states.
   c. What happens to policy iteration if the initial policy has action a in both states?
   d. Now, use value iteration.