COMP34120

Two hours

UNIVERSITY OF MANCHESTER
SCHOOL OF COMPUTER SCIENCE

AI and Games

Date: Tuesday 4th June 2013

Time: 09:45 - 11:45

Please answer any THREE Questions from the FOUR Questions provided

Use a SEPARATE answerbook for each QUESTION

This is a CLOSED book examination

The use of electronic calculators is NOT permitted

1. a) Find all pure strategy equilibrium points for this game and give its value.

(3 marks)

$$\begin{vmatrix} 1 & 1 & 2 & 1 & 1 \\ -1 & 2 & 1 & 2 & 1 \\ 0 & 1 & 1 & -1 & 0 \\ 1 & 2 & 1 & 1 & -1 \\ 1 & 1 & 1 & 1 & 1 \end{vmatrix}$$

b) Consider a 2-person zero-sum game with the following pay-off matrix:

$$\begin{vmatrix} 1 & 5 \\ 4 & 2 \end{vmatrix}$$

Describe how you would go about finding equilibria in that game. (4 marks)

c) Consider the following game played by three players: There is a prize of \$30 available. Each player can buy a ticket worth \$30, a ticket worth \$15, or no ticket at all. Once all the players have made their choice secretly the prize money is distributed evenly among those players who have bought the highest-priced ticket that was sold. If no tickets were purchased then the prize is not allocated.

Find as many equilibrium points as you can. *Hint: If you use the idea of dominance you may be able to cut down on the number of strategies.* (8 marks)

d) Under which circumstances does it make sense to speak about *the value* of a game, assuming this is a single number? How should this number be defined? Give an alternative definition that leads to the same number. (5 marks)

2. a) Assume that the game tree for a game is held in computer memory.

    i) What does the minimax algorithm calculate if the game is 2-person zero-sum? (2 marks)

    ii) What if the game is 2-person but not necessarily zero-sum? (2 marks)

    iii) Can the algorithm be adjusted to the situation where the game is *n* person? How should this be done? What does it calculate under these circumstances? (4 marks)

   b) Compare the minimax algorithm to alpha-beta pruning.

    i) What is the improvement of alpha-beta pruning over the minimax algorithm? (1 mark)

    ii) How is this achieved? (3 marks)

   c) How are these algorithms employed in a situation where the game tree does not fit into memory? Describe how they fit into a game-playing program under these circumstances. (4 marks)

   d) Pick a variation of alpha-beta pruning. Give a short description of how it works. Under which circumstances is it preferable to standard alpha-beta pruning? (4 marks)

3. a) Answer the following questions:

   i) A two-person Stackelberg game problem with perfect information can be solved by two sequential steps, in which each step solves a single player's payoff maximisation problem. Please give the mathematical description of each of these two steps. (2 marks)

   ii) For a two-person Stackelberg game with imperfect information, if the historical data about the players' past strategies are available, how should the above two steps in Question i) be revised in order to find an approximate Stackelberg strategy? (2 marks)

   b) Find a Stackelberg strategy for the following Stackelberg game:

   - There are two players in which Player L is the leader and Player F is the follower;
   - The strategy spaces for the leader and the follower are continuous ones as $U_L = [0, +\infty)$ and $U_F = [0, +\infty)$;
   - The payoff functions for the leader and the follower are

   $$J_L(u_L, u_F) = u_L(15 - \tfrac{9}{4}u_L + u_F)$$
   $$J_F(u_L, u_F) = u_F(4 + u_L - 2u_F)$$

   in which $u_L \in U_L$ is the leader's strategy and $u_F \in U_F$ is the follower's strategy.

   Note. 8 marks are for the step by step process to obtain the solution, and 1 mark is for the correct answer. (9 marks)

   c) Find a Stackelberg strategy for the following Stackelberg game:

   - There are two players in which Player L is the leader and Player F is the follower;
   - The strategy spaces for the leader and the follower are finite ones and contain only two strategies as $U_L = \{2, 3\}$ and $U_F = \{1, 2\}$ (i.e., the leader can only choose his strategy u_L either as 2 or 3 and the follower can only choose his strategy u_F either as 1 or 2);
   - The payoff functions for the leader and the follower are

   $$J_L(u_L, u_F) = (u_L - 1)(10 - 2u_L + u_F)$$
   $$J_F(u_L, u_F) = (u_F - 1)(5 + u_L - u_F)$$

   in which $u_L \in U_L$ is the leader's strategy and $u_F \in U_F$ is the follower's strategy.

   Note. 6 marks are for the step by step process to obtain the solution, and 1 mark is for the correct answer. (7 marks)

4. a) Compare reinforcement learning and supervised learning as applied to games by describing one way each is applied to learning in games. (2 marks)

   Which (reinforcement learning or supervised learning) is more commonly used in game learning and why? (2 marks)

   b) Name three heuristics you used or could have used for your Kalah bot from the Semester 1 project. Describe, using some learning method covered in this course, how your bot could use learning to discover the best heuristic (or combination of heuristics) in real time. You need not describe the algorithm in detail. (3 marks)

   c) This question concerns a game called "simplified die-roll poker", which is described below. About this game, do the following:

      i) Draw the game tree. (Don't draw each possible dice combination; treat that abstractly.) (2 marks)

      ii) Using a learning method from this course, describe an approach to learning to play this game. (6 marks)

      iii) Will your method learn to play optimally? Why or why not? (1 mark)

      Simplified die-roll poker

      A. Players alternate being player 1 and player 2.

      B. Each player puts a pound in the pot.

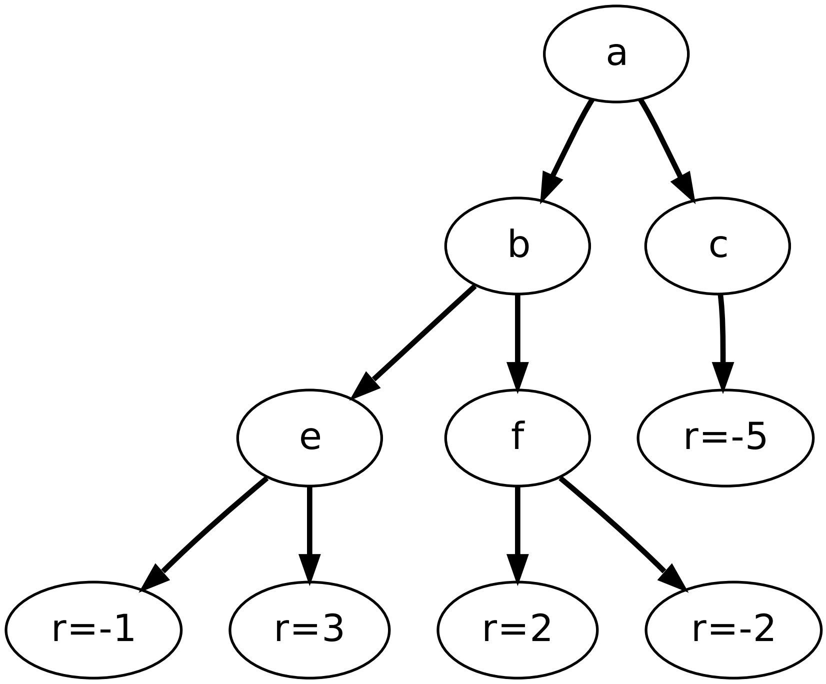
      C. Each player simultaneously rolls a hidden (from the other) die. It is an ordinary 6-sided die with the numbers 1 through 6 on the sides.

      D. Player 1 can pass or bet. If it bets, it puts 1 pound in the pot.

      E. Player 2 can call or fold. If player 2 folds, player 1 gets the pot. If player 2 calls, it has to put a 1 pound in the pot only if player 1 bet in the previous step.

      F. If player 2 did not fold in the previous step, both players show their die. High number takes the pot; in case the numbers are the same, the money stays in the pot for the next game.

   d) Consider the search tree below. Suppose Q-learning is used to learn the value of the action which can be taken from each node. Give the values for each node-action pair that Q-learning would converge to. What is the value of the top node? You want to maximise the reward *r*. (4 marks)



END OF EXAMINATION