Problem Solving (Retake Second Partial)

January 30, 2023, Classroom Q3/0003, 18:00–20:00

No notes, no communication.
The **number between brackets** [ ] indicates the points of the exercise.
Write the answer to the questions and each problem in **a different piece of paper**.
Always **reason your answer**; otherwise, it will not be considered.
Do not forget **to write your name** and coordinates (row and column) on every page.

1. In a two-player zero-sum game, the initial state is A and is the turn of player 1. The succesor states are as follows: $succ(A) = (B, C, D)$ by actions a_1, a_2, a_3; $succ(B) = (E, F, G)$ by actions b_1, b_2, b_3; $succ(C) = (H, I, J)$ by actions c_1, c_2, c_3; $succ(D) = (K, L, M)$ by actions d_1, d_2, d_3. The evaluation function $eval$ applied to states of the second level is as follows:

   $eval(E) = 0.6, eval(F) = 0.8, eval(G) = 0.4$
   $eval(H) = 0.1, eval(I) = 0.6, eval(J) = 0.4$
   $eval(K) = 0.7, eval(L) = 0.5, eval(M) = 0.6$

   (a) **[1]** If A is a MAX node and B, C, D are MIN nodes, develop the game tree and show how minimax propagates the evaluation values, identifying the best action for player 1 in A.

   (b) **[1]** Explain how to apply alpha-beta pruning and show the pruned branches.

2. For a particular zero-sum game you consider the following modification of MINIMAX with alpha-beta: when a node is going to be pruned by alpha-beta, the node is finally generated but its utility is randomly assigned (in the proper range of utility values). In this case, it is requested:

   (a) **[1]** The moves generated by this version are equal to the moves generated by the original MINIMAX with alpha-beta?

   (b) **[1]** Can alpha-beta pruning be applied to this new tree (the game tree with the new utility values)?

3. For a particular zero-sum game you consider the following modification of MINIMAX with alpha-beta: when a node is going to be pruned by alpha-beta, the node is finally generated but its utility is randomly assigned (in the proper range of utility values). In this case, it is requested:

   (a) **[1]** The moves generated by this version are equal to the moves generates by the original MINIMAX with alpha-beta?

   (b) **[1]** Can alpha-beta pruning be applied to this new tree (the game tree with the new utility values)?

4. **[2]** Translate into a CNF formula the constraint $X_1 + X_2 + X_3 + X_4 \leq 2$ using **ONE** of the methods explained in the course at your convenience (sequential counters, bubble-sort sorting networks or merge-sort sorting networks). Indicate the chosen method.

5. **[3]** Remember that 3 is the only number that is both bigger or equal to 3, and smaller or equal to 3.

   (a) Explain how to encode $X_1 + \ldots + X_n = k$ as a pair of inequalities.

   (b) Draw a merge-sorting network with 4 inputs and 4 outputs.

   (c) How many AND and OR gates does it have?

   (d) In order to use it to encode $X_1 + X_2 + X_3 + X_4 = 2$, which outputs will you use and which unitary clauses will you add to the Tseitin encoding of the sorting network to enforce the constraint? (You do not need to compute the Tseiting encoding)

   (e) If you remove useless AND and OR gates from this network, how many gates will remain?

   (f) Given $X_1 + \cdots + X_n = k$ is equivalent to $\overline{X_1} + \cdots + \overline{X_n} = n - k$. There it is any advantage in using the second form instead of the first? What if we use sequential counters? Reason your answer.