Homework 4 – Artificial Intelligence

Teacher: Stefán Ólafsson

February 14, 2023

Time Estimate

4 hours including reading the relevant parts of chapter 6 (up to 6.3) on Constraint Satisfaction Problem, assuming you have been to the lectures or watched the recordings.

Instructions

Hand in a PDF file with the answers to all questions on Canvas.

Questions

1. CSP (40 points)

Consider the problem of constructing (not solving) crossword puzzles, that is, fitting words into a rectangular grid. The grid, which is given as part of the problem, specifies which squares are blank (places for letters) and which are shaded. Assume that a list of words (i.e., a dictionary) is provided and that the task is to fill in the blank squares by using words from the list (not necessarily all of the words in the list). Assume there are n words in the list of words, you are using a 26 letter alphabet, the grid has width w and height h and the number of shaded squares on the grid is small compared to the number of blank ones.

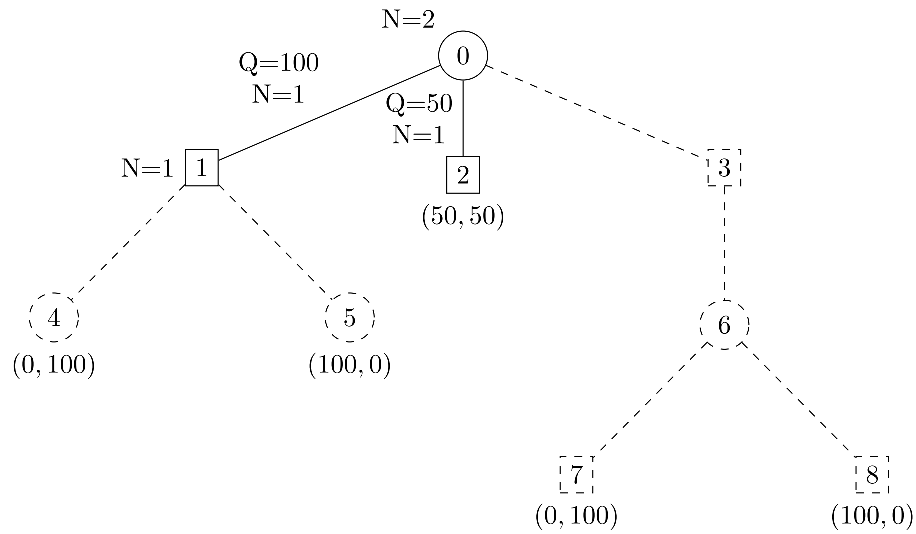
1.1 (26 points) Formulate this problem as a constraint satisfaction problem, i.e., define variables, their domains and the constraints between the variables.

1.2 (14 points) Estimate the size of the state space and the size of the search tree for the CSP given your model above. (Estimates will probably depend on w, h and n.) Explain shortly how you get to these estimates.

1.3 **(10 bonus points)** Come up with a different way of modeling this problem and shortly describe it. (Hint: Let the variables represent something different than before.) How big is the state space for this model? Is it bigger or smaller than before?

2. MCTS (60 points)

The following tree is the result of performing two iterations of Monte-Carlo tree search. The dashed lines and nodes are actions and nodes that have not been explored yet (are not in the MCTS tree), but are part of the game. The game is two-player and turn-taking. The shape of the node represents the player who is in control (circles for player 1 and rectangles for player 2). Actions are labeled with their Q and N values, nodes with their N values. In the tree below we have for example $N(0) = 2$, $Q(0, left) = 100$ and $N(0, left) = 1$. Leaf nodes belong to terminal states of the game and the values under them show both players' scores in those states. For example, in node 4, player 1 gets 0 and player 2 gets 100 points. Note that, Q values of actions are always from the perspective of the player who does the action.



UCB formula for the selection phase:

$$
\begin{aligned}
ucb(s, a) &= \begin{cases} Q(s, a) + 20 * \sqrt{\frac{ln(N(s))}{N(s,a)}}, & \text{if } N(s, a) > 0 \\ +\infty, & \text{otherwise} \end{cases} \\
\text{select } a &= \underset{a \in legals(s)}{\text{argmax}} \; ucb(s, a)
\end{aligned}
$$

2.1 (36 points) Using the tree above, perform four more iteration of MCTS/UCT (i.e., use the UCB formula above for selection). Actions with $N = 0$ are considered to have a value of $+\infty$. Select the action with the highest UCB value. If there are several actions with the same (highest) ucb value, select the left-most one. In the expansion phase, add one node to the tree – the first node on the simulation that is not in the tree yet – unless there is no such node. In the playout phase, assume that the random selection selects the left-most action. In the backpropagation phase, update the Q and N values of all actions and nodes in the simulation that are inside the tree. Keep track of the changes that happen to the tree (nodes that get added, values that change).

Note that in practice you would select randomly between equally good options, but for the purpose of this exercise, select the left-most one of equally good options, to make it easier for us to check your solution.

Write down the nodes that get added to the tree in the order in which they get added and the final Q and N values of all actions in the tree.

2.2 (6 points) Which move would player 1 select in state 0 according to the MCTS tree after these iterations?

2.3 (6 points) Which move is the optimal move (rational choice) for player 1 in state 0 assuming a rational opponent?

2.4 (12 points) Would MCTS eventually find the optimal move given enough iterations? Explain.

2.5 **(10 bonus points)** Using this game as an example, explain why the exploration part in the UCB formula is important for finding the optimal move (What would happen with C=0?).

2.6 **(10 bonus points)** Derive an estimate of how many more iterations are needed until MCTS would show the optimal move for player 1 in state 0, that is, until the optimal move is the one with the highest Q.