

1. What heuristic did you use? Why?

The heuristic used is an evaluation function that is directly proportional to the likelihood of winning at every given state of the board. The expectimax score is calculated by multiplying probabilities of event occurrences with heuristic evaluation scores.

2. Describe how your algorithm performs given different time constraints. How much of the tree can you explore given 5 seconds per turn? 10 seconds? 3 seconds?

The algorithms implemented in this game are depth-limited searches, so with increase in the number of seconds, more nodes of the tree can be explored. Hence, the algorithm makes better decisions when 10 seconds per turn, as it gets to explore more nodes and decide the best move. When the time per turn is reduced to 3 seconds, the algorithm has very limited time to expand and search the tree, so the result decision given by the algorithm might not be the best.

Thus the number of nodes of the tree explored by the algorithm will be in the order: 10 seconds > 5 seconds > 3 seconds.

3. Can you beat your algorithm?

Yes, Connect Four is a strongly solved game which means that the first player can always win the game provided they make the optimal moves, irrespective of who the opponent is. So I can best my algorithm by making the right moves.

4. If your algorithm plays itself, does the player that goes first do better or worse in general? Share some of the results.

If the algorithm plays itself, the player who goes first will do better provided they start in the center column. The game will be a draw if the player starts on either side. The game will be a loss for the first player if they start on the outer 2 columns on either side.

Controlling the center column is the best way to win. So if the first player makes the first move in the side column and the second player makes the first move in the center column, then the second player will have higher chances of winning the game.