Build a Isolation Game Playing Agent
AI Nanodegree Project Report

Vishakha Sharma

February 26, 2017

1 Introduction

In this project, we develop an AI agent for playing the game of Isolation. Isolation is a two-player game played on a rectangular grid of positions where both players take turns to move their piece and the first player who runs out of available moves loses the game. The positions occupied in previous moves are blocked for all future moves and in this specific version of Isolation game, the pieces are only allowed to move in L-shaped steps, just like a knight on a Chess board.

The main objectives of this project are:

- Implement Minimax algorithm for searching the game tree.
- Implement Alpha-Beta pruning to improve the efficiency of the minimax game tree search.
- Utilize Iterative Deepening (ID) along with the Alpha-Beta search to implement the Isolation game AI agent that returns the best next move within the time provided to it.
- Develop heuristic evaluation functions that perform comparably or better than the provided heuristic evaluation functions.

2 Heuristic Evaluation Functions

We propose three different heuristics in addition to the ones provided in the starter package for the project. The heuristic evaluation functions provided to us are described as follows:

- Null Score: This function returns 0 for all game board states which are non-terminal states.
- Open Moves Score: This function returns the number of moves available to player in a given board state as the score for non-terminal states.

- Difference of Open Moves Score (Improved): This function returns the difference in the number of moves available to the player and the opponent player in a given board state as the score for non-terminal states.

The proposed heuristic evaluation functions are described as follows:

- Weighted Open Moves (H1): The idea here is to modify the Open moves score heuristic provided to us by weighting each open move by a weight that depends on the position the move leads to on the game board. The motivation is that positions in the center of the board provide more open moves than those near the edges. The weight for a position is set as the maximum number of moves available from that position on the board as shown below for a 7×7 board.

2	3	4	4	4	3	2
3	4	6	6	6	4	3
4	6	8	8	8	6	4
4	6	8	8	8	6	4
4	6	8	8	8	6	4
3	4	6	6	6	4	3
2	3	4	4	4	3	2

- Difference in Weighted Open Moves (H2): In this heuristic, we calculate the difference in the weighted open moves scores (H1) between the current player and its opponent and use that as the score of the current game state.

- Difference in Open Moves One Ply Ahead (H3): The difference in the number of available moves between the current player and its opponent one ply ahead in the future is used as the score of the current game state.

3 Results

The performance of the Isolation playing AI agents is evaluated using a tournament setup which consists of a Random AI agent that moves randomly and a set of AI agents utilizing minimax search (MM) and minimax with alpha-beta pruning (AB) search algorithms with fixed depth game trees as the opponents which use the Null, Open, Improved heuristic evaluation functions and the players includes agents that use iterative deepening (ID) with the Improved, H1, H2 and H3 heuristic evaluation functions.

To estimate and compare the performance of the proposed heuristic evaluation functions to the ID-Improved agent, we simulate 10 tournaments each consisting of 20 matches against every opponent for a particular heuristic evaluation function and another longer tournament consisting of 400 matches per opponent for a particular heuristic evaluation function as shown in Table 1 and

Table 2 respectively. We observe that the ID-H3 player consistently performs better than all the other heuristics considered. The gain in performance is around 5%.

Table 1: Table showing the results for all the ID agents with different heuristics in 10 tournaments each consisting of 20 matches per opponent.

Tournament No.	ID-Improved	ID-H1	ID-H2	ID-H3
1	76.43%	75.00%	82.14%	86.43%
2	74.29%	74.29%	82.14%	76.43%
3	77.14%	77.14%	75.71%	78.57%
4	81.43%	77.86%	77.86%	79.29%
5	77.86%	76.43%	77.14%	79.29%
6	82.86%	73.57%	77.14%	85.00%
7	76.43%	74.29%	79.29%	79.29%
8	77.14%	78.57%	76.43%	77.86%
9	74.29%	75.00%	78.57%	80.71%
10	80.00%	74.29%	81.43%	77.86%
Mean	77.79%	75.64%	78.79%	80.07%
Std. Dev.	2.70%	1.64%	2.26%	3.04%

Table 2: Table showing the results for all the ID agents with different heuristics in a tournament consisting of 400 matches per opponent.

	Random	MM-Null	MM-Open	MM-Improved	AB-Null	AB-Open	AB-Improved	Win %
ID-Improved	369 to 31	348 to 52	292 to 108	284 to 116	322 to 78	269 to 131	262 to 138	76.64%
ID-H1	377 to 23	344 to 56	303 to 97	278 to 122	326 to 74	287 to 113	269 to 131	78.00%
ID-H2	382 to 18	356 to 44	306 to 94	317 to 83	316 to 84	290 to 110	277 to 123	80.14%
ID-H3	372 to 28	362 to 38	333 to 67	315 to 85	325 to 75	291 to 109	281 to 119	81.39%

The H3 heuristic evaluation function is recommended to be used and can be justified as follows:

- The H3 heuristic is based on the number of moves looking ahead one ply in the future. This is important in this version of isolation where only L-shaped knight like moves are allowed. Due to these kinds of moves, its difficult to predict the value of a game state by just counting the number of immediately available moves like in the Improved heuristic. Hence one ply lookahead should provide a more accurate evaluation function.

- The H3 heuristic running time complexity is comparable to the Improved heuristic and hence it should not adversely affect the maximum depth searched.

- As noted from the results, the H3 heuristic does perform better in practice and hence can be recommended.