AI agent plays Isolation Game

Jean-Marc Beaujour,
AI Nanodegree Project Report

May 26, 2017

Abstract

We develop an AI agent for playing the game of Isolation. The game consists of a board and 2 players who take turn in filling up the board/grid, where the positions occupied in previous moves are not available for all future moves. The goal is to prevent the opponent any more moves.

1 Introduction

To provide the AI agent with the best next move, we implement:

1. a minimax algorithm for searching the game tree

2. alpha-beta pruning to improve the efficiency of tree search

3. Iterative Deepening to improve the time efficiency

4. custom heuristic evaluation function to further improve the AI performance.

Several Heuristic are proposed and their performance are compared to that of 'ID improved'.

2 Custom Heuristic Evaluation Functions

Four Heuristic evaluation functions (f_{eval}) are evaluated. The functions represent different ways of expressing the difference between the number of moves available for the AI agent (N_{AI}) and that of the opponent N_{opp}.

2.1 Heuristic 1

- Mathematical expression:

$$f_{eval} = \frac{N_{AI}}{N_{opp}} \tag{1}$$

The evaluation function computes the ratio of the available moves of the player over the number of moves of the opponents. A large (a small) f_{eval}, i.e $N_{AI} >> N_{opp}$ ($N_{AI} << N_{opp}$) favors a win (loss) for the player. Nevertheless, at the beginning of the game, f_{eval} is likely to be in the lower range (≈ 1), because $N_{AI} \approx N_{opp}$.

Match	opponent	ID_Improved wins	Student wins	Δ
1	Random	19 to 1	19 to 1	0
2	MM_Null	14 to 6	15 to 5	+1
3	MM_Open	14 to 6	12 to 8	-2
4	MM_Improved	10 to 10	13 to 7	+3
5	AB_Null	10 to 6	12 to 8	+2
6	AB_Open	12 to 8	8 to 12	-4
7	AB_Improved	9 to 11	11 to 9	+2
average		62.86%	64.29%	+ 1.43%

2.2 Heuristic 2

- Mathematical expression:

$$f = \frac{N_{AI}}{N_{AI} + Nopp} \tag{2}$$

The evaluation function computes the percentage of total moves available for the player. In contrast to the previous Heuristic, here the evaluation function remains in the range $[0, 1]$. If $f_{eval} \rightarrow 0$, the opponent is likely to win, and if $f_{eval} \rightarrow 1$, the player is likely to win.

Match	opponent	ID_Improved wins	Student wins	Δ
1	Random	18 to 2	19 to 1	+1
2	MM_Null	18 to 2	17 to 3	-1
3	MM_Open	7 to 13	8 to 12	+1
4	MM_Improved	10 to 10	10 to 10	0
5	AB_Null	13 to 7	14 to 6	+1
6	AB_Open	9 to 11	12 to 8	+3
7	AB_Improved	9 to 11	9 to 11	0
average		60.00%	63.57%	+3.57%

2.3   Heuristic 3

- Mathematical expression:

$$f_{eval} = \frac{N_{AI} - N_{opp}}{N_{opp}} \tag{3}$$

The evaluation function computes the difference between the moves available for the player and the opponent available moves, and normalize the difference by number of opponent moves. When $N_{AI} >> N_{opp}$, then $f_{e}val$ is large and positive. In contrast, if $N_{AI} << N_{opp}$, $f_{e}val$ is small and negative.

Match	opponent	ID_Improved wins	Student wins	Δ
1	Random	18 to 2	18 to 2	0
2	MM_Null	17 to 3	18 to 2	+1
3	MM_Open	13 to 7	9 to 11	-4
4	MM_Improved	10 to 10	13 to 7	+3
5	AB_Null	15 to 5	10 to 10	-5
6	AB_Open	9 to 11	10 to 10	+1
7	AB_Improved	10 to 10	9 to 11	-1
average		65.71%	62.14%	-3.57%

2.4   Heuristic 4

- Mathematical expression:

$$f_{eval} = \frac{2 * N_{AI} - N_{opp}}{N_{opp}} \tag{4}$$

3

In this function, we calculate the weighted difference between the number of available moves for the player vs. that of the opponent. This value is normalized to the number of moves of the opponent. In the numerator, the player's moves are set to be worth 2 times more than the opponent moves.

Match	opponent	ID_Improved wins	Student wins	Δ
1	Random	20 to 0	19 to 1	+1
2	MM_Null	16 to 4	12 to 8	-4
3	MM_Open	12 to 8	9 to 11	-3
4	MM_Improved	10 to 10	12 to 8	+2
5	AB_Null	14 to 6	16 to 4	+2
6	AB_Open	12 to 8	5 to 15	-7
7	AB_Improved	9 to 11	9 to 11	0
average		66.43%	58.57%	-7.86%

2.5   Analysis

The best evaluation function is Heuristic 2 : it outperforms the 'ID_improved' model by 3.57 points. Interestingly, the evaluation function H1 is also better than 'ID_improved' but with a smaller margin. This would suggest that it is more important to track the percentage of moves available for the player among all the available moves.