Game Playing Agent Heuristic Analysis

Carlos Aguilar

Udacity Artificial Intelligence Nanodegree

March 2017

Introduction

This document presents a list of heuristics used to play an Isolation game variant on a computer agent. It Also presents statistics about the heuristic's performance and an analysis on why they perform as they do.

Heuristics

Open move score (same as the lecture)

This heuristic simply counts the number of available moves for the player on the current state and assigns the count as the heuristic score.

Improved score (same as the lecture)

This heuristic calculates the difference between the number of available moves for the current player and the number of available moves for the opposite player

Weighted improved score

This heuristic uses the same principle than the improve score but it gives more relevance to how many moves the current player has available. This is done multiplying the available moves by a factoring number. After several tests, it was determined that the best balance is that the player's moves has double score than the opponent's moves.

Possible future moves evaluation

This heuristic is somehow more complex than the previous ones because it uses the knowledge that if some cells are empty it means that it is possible to move more than once after moving to the current cell.

The heuristic uses a grid as the following:

16	1	4	1	16
1	4	8	4	1
4	8	P	8	4
1	4	8	4	1
16	1	4	1	16

Where P is the player and let's say it is at (0, 0) so this heuristic evaluates al possible moves for P, and assigns a score depending on how far can the movement go further on its path. For example, if (-1, -2) is empty it assigns 1 score point to the move as we can see in Figure 1, we know that if we are in (-1, -2) the only move we can do on this 5 x 5 grid is to (0, -2) or (1, -1) (Figure 2), so for each empty cell in this list it adds 4 points to the score (because it means it can move up to two times if that move is selected). Then for each empty cell it determines if there are moves left on that path, if we choose (0, -2) then it validates if (-1, 0) and (1, 0) are empty (Figure 3), if any of those are empty it will add 8 to the score and evaluates the final step for each of them. Finally, for the case of (-1, 0) the heuristic now evaluates if (-2, -2) and (-2, 2) are empty (Figure 4) for each empty cell in this list it adds 16 points to the score.

Figure 1

16	1	4	1	16
1	4	8	4	1
4	8	P	8	4
1	4	8	4	1
16	1	4	1	16

Figure 2

16	1	4	1	16
1	4	8	4	1
4	8	P	8	4
1	4	8	4	1
16	1	4	1	16

Figure 3

16	1	4	1	16
1	4	8	4	1
4	8	P	8	4
1	4	8	4	1
16	1	4	1	16

Figure 4

16	1	4	1	16
1	4	8	4	1
4	8	P	8	4
1	4	8	4	1
16	1	4	1	16

This means that the heuristic will score higher when there are more moves left to follow on a given move.

Heuristic Analysis

To analyze the heuristics, I used the tournament.py file provided with the project to run several times the agents using different heuristics. For comparison, I didn't evaluate the open score heuristic. I ran each heuristic 10 times and the average results are presented on the following table:

Improved Score	Weighted Improved Score	Possible Future Values
************************	************************	************************
Evaluating: ID_Improved	Evaluating: Student	Evaluating: Student
************************	************************	************************
Playing Matches:	Playing Matches:	Playing Matches:
----------	----------	----------
ID_Improved vs Random Result: 18 to 2	Student vs Random Result: 18 to 2	Student vs Random Result: 19 to 1
ID_Improved vs MM_Null Result: 16 to 4	Student vs MM_Null Result: 14 to 6	Student vs MM_Null Result: 18 to 2
ID_Improved vs MM_Open Result: 6 to 14	Student vs MM_Open Result: 13 to 7	Student vs MM_Open Result: 13 to 7
ID_Improved vs MM_Improved Result: 15 to 5	Student vs MM_Improved Result: 17 to 3	Student vs MM_Improved Result: 17 to 3
ID_Improved vs AB_Null Result: 20 to 0	Student vs AB_Null Result: 20 to 0	Student vs AB_Null Result: 20 to 0
ID_Improved vs AB_Open Result: 20 to 0	Student vs AB_Open Result: 20 to 0	Student vs AB_Open Result: 20 to 0
ID_Improved vs AB_Improved Result: 20 to 0	Student vs AB_Improved Result: 20 to 0	Student vs AB_Improved Result: 20 to 0
Results:	Results:	Results:
----------	----------	----------
ID_Improved 82.14%	Student 87.14%	Student 90.71%

It seems that the improved score is better than the random strategy but it might fail against the open heuristic, this might be because on this isolation's variant the opponent moves are not as important as actually how many moves you have left. Also, this might also explain why giving a higher weight to the player's move over the opponent's move also results on a better approach.

Finally, the last heuristic seems to have the best result, which is the expected since this heuristic not only uses the knowledge of how many moves can you make on the following step but it also analyzes how many moves afterwards can be performed. This is better because with the current rules once you move the amount of moves you could performed on the previous steps are irrelevant on your current position while the final heuristic recognizes that after one move (if the opponent does not block the player) it still have n -1 possible moves left than the previous step so if there are no other better movement it will follow the path and is there is it will have a new path to follow.