Udacity AIND Isolation Game heuristics Analysis

1. Background

In this project, we needed to devise heuristics that can outperform an agent(AB_Improved) with alpha-beta pruning, iterative deepening search and a heuristic function of "improved". This heuristic scores a board as `(active_player_move_count - active_player_move_count)`

Usually there is a trade off between a complex and more accurate heuristic vs simple but fast heuristic. When you have limited time to search through layers of board positions, a slow but accurate heuristic can reduce the depth of search before it times out. While a fast and simple heuristic can search more depths. And there is no correct answer. It is a delicate balance.

So in order to outdo "improved" heuristic, I decided to use two extreme ends of heuristics. First set being the ones that are simpler than "improved" and others that are more complex as compared to "improved" heuristic. Given below the is the list of final 3 choices I worked with.

2. Heuristics chosen and their relative performance

Details of three heuristics are give below

Heuristic_1 (AB_Custom): A way to penalise moves into corner along with improved by combining "improved" and "center_score" heuristics. The theoretical range of heuristic for "improved" (own-oppo moves) is [-8,8], while the max for center_score for a 7x7 board is $2*3^2 = 18$ i.e. a range of [0,18]. The overall range of both are in comparable and hence can be combined easily together without needing any scaling as given below. We subtract "center_score" as this heuristic has a zero value at the center and it increases as position moves to corners.

```
final_score = w * improved_score – (1-w) * "center_score"
```

The weight used to combine was chosen as 0.5 for both scores. However, a better way to adjust the weights would be by tuning it over multiple simulations for different values of weights.

Heristic_2 (AB_Custom_1): Just return -ve of open moves for the opponent. This is opposite of the `open_move_score` heuristic in `sample_players.py`. This heuristic is a tad bit simpler and faster then "improved" heuristic that we need to beat.

Heuristic_3 (AB_Custom_2): Similar to "improved", difference between player and opponent open moves with additional penalties if the move takes active player to corners of the board. The rationale being that moving to corner can push active player to have lesser options in subsequent moves.

`tournament.py` program was run with these three heuristics for two times. Scores are reported below:

This script evaluates the performance of the custom_score evaluation function against a baseline agent using alpha-beta search and iterative deepening (ID) called `AB_Improved`. The three `AB_Custom` agents use ID and alpha-beta search with the custom_score functions defined in game_agent.py.

```
                         *************************
                              Playing Matches
                         *************************
```

Match #	Opponent	AB_Improved Won	AB_Improved Lost	AB_Custom Won	AB_Custom Lost	AB_Custom_2 Won	AB_Custom_2 Lost	AB_Custom_3 Won	AB_Custom_3 Lost
1	Random	7	3	9	1	9	1	8	2
2	MM_Open	2	8	2	8	1	9	6	4
3	MM_Center	7	3	5	5	5	5	5	5
4	MM_Improved	4	6	5	5	1	9	2	8
5	AB_Open	4	6	4	6	6	4	6	4
6	AB_Center	6	4	6	4	6	4	4	6
7	AB_Improved	6	4	3	7	4	6	4	6
	Win Rate:	51.4%		48.6%		45.7%		50.0%	

2nd run produced these metrics

Match #	Opponent	AB_Improved Won	AB_Improved Lost	AB_Custom Won	AB_Custom Lost	AB_Custom_2 Won	AB_Custom_2 Lost	AB_Custom_3 Won	AB_Custom_3 Lost
1	Random	8	2	9	1	3	7	6	4
2	MM_Open	4	6	4	6	2	8	3	7
3	MM_Center	4	6	8	2	5	5	3	7
4	MM_Improved	4	6	4	6	3	7	2	8
5	AB_Open	6	4	6	4	6	4	5	5
6	AB_Center	7	3	7	3	6	4	5	5
7	AB_Improved	5	5	6	4	5	5	5	5
	Win Rate:	54.3%		62.9%		42.9%		41.4%	

From above two runs of the tournament, we can see that there is no definitive answer on which of the three custom heuristics perform better than AB_Improved. They all seem to be in the same range of performance metric as AB_Improved.

In order to assess the impact of heuristic complexity on the search depth, I ran AB_Improved vs three customer heuristics and recorded the average depth traversed in each case. The findings are given below:

	AB_Improved	AB_custom	AB_custom_2	AB_custom_3
Initial moves	7-12	7-11	7-15	7-14
Towards end	1100-2500	1000-2500	1900-2200	1100-2400

While more complex heuristic like AB_custom seem to have search depth which is a bit lesser then custom2 and custom3, there is no appreciable pattern. It warrants more exploration. It may point to the fact that my “custom” heuristic while complex was still not complex enough over “improved” to significantly impact search depth.

3. Final Heuristic Chosen

I decided to go ahead with combined heuristic of "improved" and "center_score" with uniform weight of 0.5 to both.

The reason for this choice were:

a) It combines two concepts of difference in open moves between player and opponent with offset from center.
b) It performed as well as Improved across two runs which was better than the combined performance over two runs for custom_2 and custom_3 heuristics.
c) While it added richness to the evaluation function, it did not significantly impact the search depth as compared to "improved"
d) The weights could be further tuned to get be a better score.

4. Additional Points

The fact that Minimax based players see to perform fairly good points to some issue that AB search with ID is not offering significant benefit. This was counter intuitive and needs more investigation esp when the minimum search depth explored under alpha_beta with ID was 7.

I rechecked the logic of alpha beta and Iterative deepening to find if there were possible sources of errors but could not find one. The implemented code also passed the auto grader.

Two points need to be investigated more in depth:

a) Is the sample size of 10-20 games too low statistically to more conclusions on the effectiveness of heuristics and search methods?
b) Are there some specific opening moves which are so favoured that no matter how efficient be the search, player with those opening moves is almost guaranteed to win?
c) Are the heuristics effective in measuring the final outcome?