

Heuristic Analysis

Presented here is the analysis of the three custom heuristics for the Isolation game agent. The performance of each heuristic was gauged by running the tournament.py script. This ran a set number of games against Udacity's sample heuristics using both a fixed depth minimax algorithm and an alphabeta with pruning algorithm. A timeout of 150 milliseconds and a batch of 50 games per player was run. The improved_score heuristic was the best performer out of the sample heuristics and was used as a benchmark against which the custom heuristics were measured against. The improved_score heuristic had an overall performance score of **68%**. Shown as AB_Improved in the data.

custom_heuristic_3: Stay close to center

It was observed that most isolation games ended with the losing player trapped on the edge of the board. A heuristic was designed so that moves that placed the player further away from the center of the board were heavily penalized. This would ensure that the player stayed away from the edges as much as possible. This heuristic had an overall performance score of **67%**. Shown as AB_Custom_3 in the data.

custom_heuristic_2: Restrict opponent

The improved_score heuristic was very hard to beat. This heuristic measured the difference in the player's available moves (own_moves) vs the opponent's available moves(opp_moves). It is posited here that a move that not only increased own_moves but also decreased opp_moves should be superior. For example a move that would result in (own_moves,opp_moves) of (2,1) is superior to one that would result in (4,3) since the opponent with only 1 move left would lose the very next turn. However, improved_score would score both moves equally. It is proposed that the ratio own_moves/opp_moves would correctly score the relative strength of the above two moves. A heuristic that returned the ratio own_moves/opp_moves was proposed as a suitable heuristic. This heuristic had an overall performance score of **70%**. Shown as AB_Custom_2 in the data.

custom_heuristic: Stay close to center while restricting opponent

The above two heuristics are rather restrictive in their own in that they seem to overlook superior moves by enforcing a single strategy. A third heuristic was tested which was a combination of the above two custom heuristics. The idea behind this is that whilst we want to stay away from the edges it is important to restrict the opponent's moves too. An equal weighted sum of the two heuristics was used. This heuristic had an overall performance score of **72%**. Shown as AB_Custom in the data.

Conclusion

The heuristic, *custom_heuristic "Stay close to center while restricting opponent"* was chosen as the heuristic for game_agent. The reasons for this are:

[1] It was consistently the best performer out of the three, in numerous trials that were conducted.

[2] Computationally simple, it allowed more time to be allocated to searching more levels. This is especially important early in the game when it isn't clear what path the game is going to take. There is little benefit in allocating more time to the heuristic calculations at the expense of searching deeper.

[3] By changing the weights of the component factors *"Stay close to center"* and *"Restrict opponent"* it may be possible to have a dynamic heuristic function that can be adjusted depending on what stage of the game we are in. For example, in the early stages it may not be too important to stay in the center but it may be more critical as the number of squares available shrink in the latter stages of the game so as not to be trapped on the edges. The weights can be made dependent on the current 'stage' of the game such as beginning, middle or ending to improve the performance even further.

Performance Data and Chart:

Match	Opponent	AB_Improved		AB_Custom		AB_Custom_2		AB_Custom_3	
		Won	Lost	Won	Lost	Won	Lost	Won	Lost
1	random	96	4	97	3	94	6	92	8
2	MM_Open	75	25	78	22	76	24	69	31
3	MM_Center	88	12	91	9	88	12	83	17
4	MM_Improved	68	32	73	27	73	27	73	27
5	AB_Open	52	48	48	52	51	49	45	55
6	AB_Center	54	46	65	35	61	39	59	41
7	AB_Improved	43	57	51	49	49	51	47	53
Win Rate		68%		72%		70%		67%	

