Udacity Artificial Intelligence Project2 : Build a Game-Playing Agent

Part 1: Heuristic Analysis

Julianne Hong

Heuristic 1 (custom_score_2): The distance from current opponent's position

Heuristic 2 (custom_score_3): number of my available moves – 2*number of opponent's move

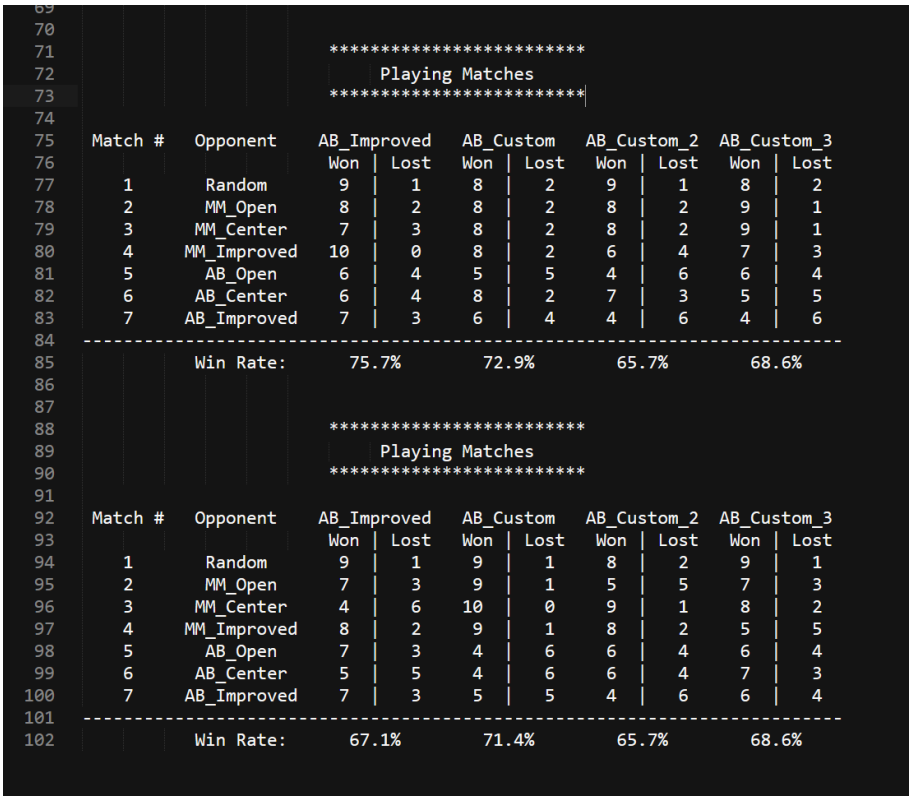
Heuristic 3 (custom_score_1): Heuristic 1 + Heuristic 2

Match #	Opponent	AB_Improved		AB_Custom		AB_Custom_2		AB_Custom_3	
		Won	Lost	Won	Lost	Won	Lost	Won	Lost
1	Random	8	2	8	2	9	1	10	0
2	MM_Open	7	3	7	3	5	5	8	2
3	MM_Center	7	3	9	1	7	3	9	1
4	MM_Improved	6	4	8	2	6	4	6	4
5	AB_Open	5	5	9	1	5	5	6	4
6	AB_Center	5	5	8	2	6	4	6	4
7	AB_Improved	5	5	4	6	7	3	4	6
	Win Rate:	61.4%		75.7%		64.3%		70.0%	

Match #	Opponent	AB_Improved		AB_Custom		AB_Custom_2		AB_Custom_3	
		Won	Lost	Won	Lost	Won	Lost	Won	Lost
1	Random	5	5	8	2	7	3	9	1
2	MM_Open	7	3	6	4	7	3	7	3
3	MM_Center	6	4	10	0	7	3	7	3
4	MM_Improved	3	7	5	5	4	6	6	4
5	AB_Open	6	4	7	3	5	5	7	3
6	AB_Center	5	5	4	6	4	6	7	3
7	AB_Improved	4	6	5	5	5	5	5	5
	Win Rate:	51.4%		64.3%		55.7%		68.6%	

Match #	Opponent	AB_Improved		AB_Custom		AB_Custom_2		AB_Custom_3	
		Won	Lost	Won	Lost	Won	Lost	Won	Lost
1	Random	7	3	9	1	9	1	8	2
2	MM_Open	6	4	8	2	4	6	7	3
3	MM_Center	7	3	9	1	7	3	9	1
4	MM_Improved	4	6	5	5	5	5	5	5
5	AB_Open	5	5	6	4	5	5	6	4
6	AB_Center	4	6	7	3	6	4	6	4
7	AB_Improved	5	5	4	6	4	6	6	4
	Win Rate:	54.3%		68.6%		57.1%		67.1%	

Figure 1: Search Depth is 3



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

Match #   Opponent    AB_Improved   AB_Custom   AB_Custom_2  AB_Custom_3
                       Won | Lost   Won | Lost   Won | Lost   Won | Lost
    1       Random      9  |  1     8  |  2     9  |  1     8  |  2
    2       MM_Open     8  |  2     8  |  2     8  |  2     9  |  1
    3      MM_Center    7  |  3     8  |  2     8  |  2     9  |  1
    4     MM_Improved  10  |  0     8  |  2     6  |  4     7  |  3
    5       AB_Open     6  |  4     5  |  5     4  |  6     6  |  4
    6      AB_Center    6  |  4     8  |  2     7  |  3     5  |  5
    7     AB_Improved   7  |  3     6  |  4     4  |  6     4  |  6
--------------------------------------------------------------------------
          Win Rate:      75.7%        72.9%        65.7%        68.6%


                        ************************
                             Playing Matches
                        ************************

Match #   Opponent    AB_Improved   AB_Custom   AB_Custom_2  AB_Custom_3
                       Won | Lost   Won | Lost   Won | Lost   Won | Lost
    1       Random      9  |  1     9  |  1     8  |  2     9  |  1
    2       MM_Open     7  |  3     9  |  1     5  |  5     7  |  3
    3      MM_Center    4  |  6    10  |  0     9  |  1     8  |  2
    4     MM_Improved   8  |  2     9  |  1     8  |  2     5  |  5
    5       AB_Open     7  |  3     4  |  6     6  |  4     6  |  4
    6      AB_Center    5  |  5     4  |  6     6  |  4     7  |  3
    7     AB_Improved   7  |  3     5  |  5     4  |  6     6  |  4
--------------------------------------------------------------------------
          Win Rate:      67.1%        71.4%        65.7%        68.6%
```

Figure 2: Search Depth is 5

All of the heuristics have fairly high winning rate against minimax players. Comparing the heuristic 1 (AB_Custom_2), that is the distance from the current opponent's position and the heuristic 2 (AB_Custom_3), that is the my move left subtract by doubled opponent's move left, the second one seems more improving than the first one. Intuitively, it makes sense because by the agent moving far away from the opponent, it is benefiting the opponent as well because the agent plays the game passively running away from it rather than aggressively attacking the opponent. Also, the first heuristic might take longer to compute.

Although the third heuristic (AB_Custom_1) which is the combination of the other two heuristics, doesn't seem to be doing much better than the other heuristics alone when search depth is 3. It is because it simply takes much longer to compute compared to the other heuristics. This can be seen by comparing Figure 1 and Figure 2. When search depth is 3 the winning rate of the heuristic 3 is as high as the other two, or sometimes even lower, even though the two evaluation functions are combined. However, when the search depth is 5, the its winning rate is significantly (well, compare to the depth 3) higher than the other two. Considering the tournaments ran on a standard machine, which can affect the game score by time constraints, I can conclude that heuristic 3 would be the best option to go if any of these three heuristics would be used in a real game tournament.