Heuristic Analysis for Isolation Game

1. Custom Heuristic Score Functions Implemented

1.1 custom_score(): `squared_num_moves_diff_score()`

This heuristic computes the difference between the squared number of legal moves for the player and the squared number of moves for the opponent.

This heuristic has the effect of accentuating the difference in the number of moves, compared to `improved_score()`, hence a more rewarding or more penalizing differential effect, even for the same numbers of moves (e.g., 5-4 = 1, but 5x5-4x4=9). Whereas `improved_score()` will return the same score for the same net difference (e.g., 5-4=1, 4-3=1, 3-2=1, etc.), `squared_num_moves_diff_score()` will return a different score in each case (e.g., 5x5-4x4=9, 4x4-3x3=7, 3x3-2x2=5, etc.), favoring a higher number of moves available for the player, even if the opponent has a similar number of moves available.

1.2 custom_score_2(): `one_step_look_ahead_score()`

This heuristic forecasts each of the legal moves for either the given player or its opponent, depending on who is the active player at the given game state, calculates weighted number of moves according to the next game state that the move leads to, and returns the difference between weighted average numbers of moves as the score.

In case the player is the active player, the heuristic calculates the weighted number of the player's own moves, according to how good or bad the next state would be for the player, depending on each of the legal moves available for the player. In case the player is the inactive player, the heuristic calculates the weighted number of the opponent's moves, according to how favorable or unfavorable the next state would be for the opponent, depending on each of the legal moves available for the opponent.

Like `squared_num_moves_diff_score()`, this heuristic can also be considered as a variation based on `improved_score()`. Whereas `improved_score()` function treats each legal move, either for the player or the opponent, as of equal weight, i.e., as unit 1, and takes the difference between the sums of moves, `one_step_look_ahead_score()` treats each move differentially, according to the predicted favorability or unfavorability of the game state that follows from each move.

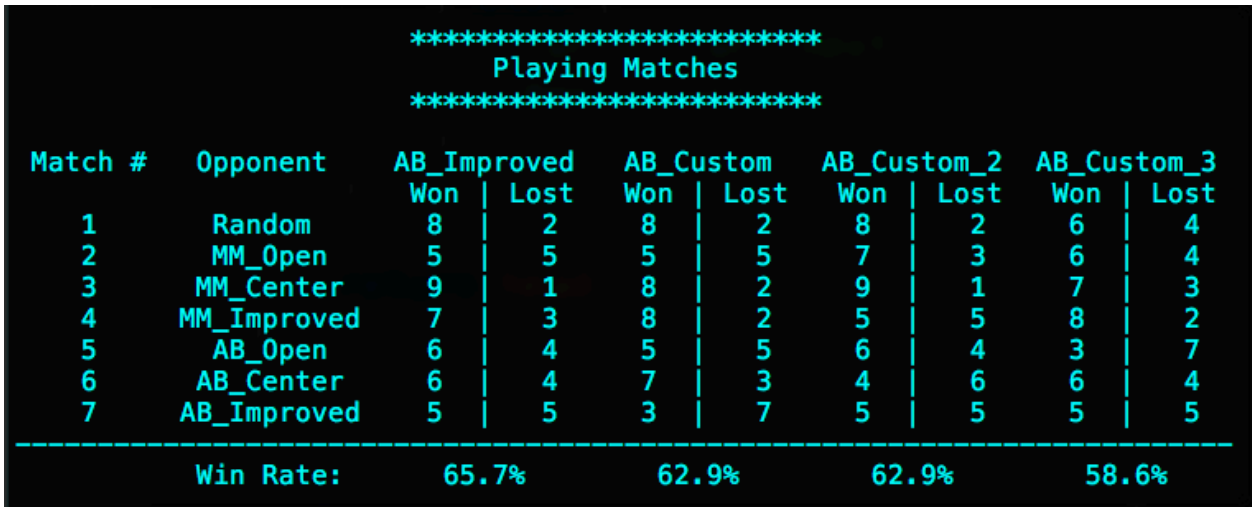
1.3 custom_score_3(): `num_moves_ratio_score()`

This heuristic computes the ratio of the number of legal moves for the player to the number of legal moves for the opponent.

In contrast to `squared_num_moves_diff_score()`, this heuristic will put a higher weight on the difference in the number of moves when the numbers are lower (e.g., 2.0/1.0 = 2 vs. 5.0/4.0 = 1.25).

2. Experiment Setup and Results

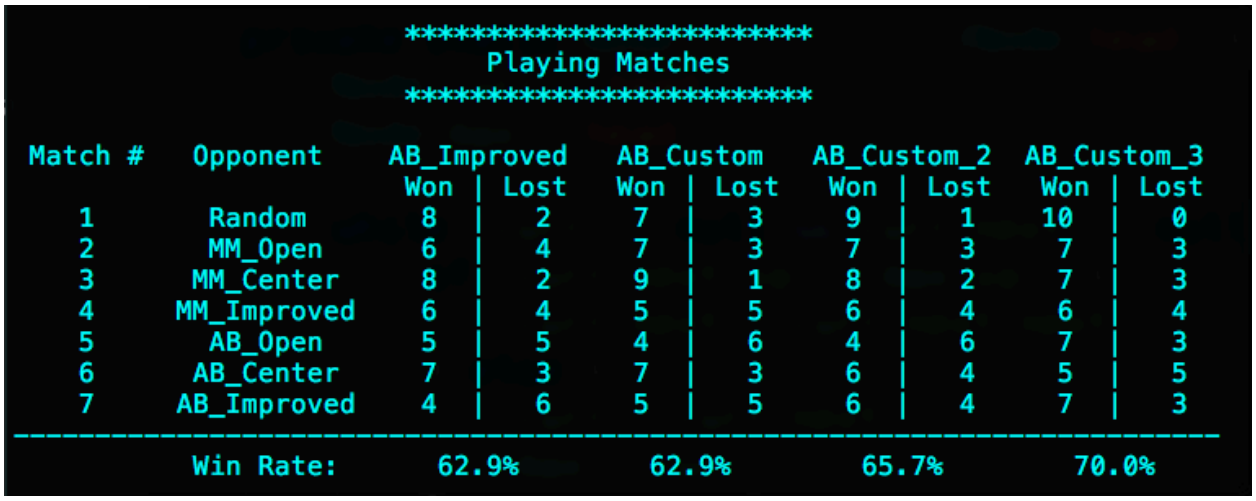
The above 3 custom heuristics have been tested in comparison to the base agent using `improved_score()`, against 7 opponents, across multiple tournaments. Figs. 1–10 show the result of each tournament with 10 games each per opponent.



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

 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      8  |   2      6  |   4
    2         MM_Open       5  |   5      5  |   5      7  |   3      6  |   4
    3        MM_Center      9  |   1      8  |   2      9  |   1      7  |   3
    4       MM_Improved     7  |   3      8  |   2      5  |   5      8  |   2
    5         AB_Open       6  |   4      5  |   5      6  |   4      3  |   7
    6        AB_Center      6  |   4      7  |   3      4  |   6      6  |   4
    7       AB_Improved     5  |   5      3  |   7      5  |   5      5  |   5
--------------------------------------------------------------------------------
           Win Rate:         65.7%         62.9%         62.9%         58.6%
```

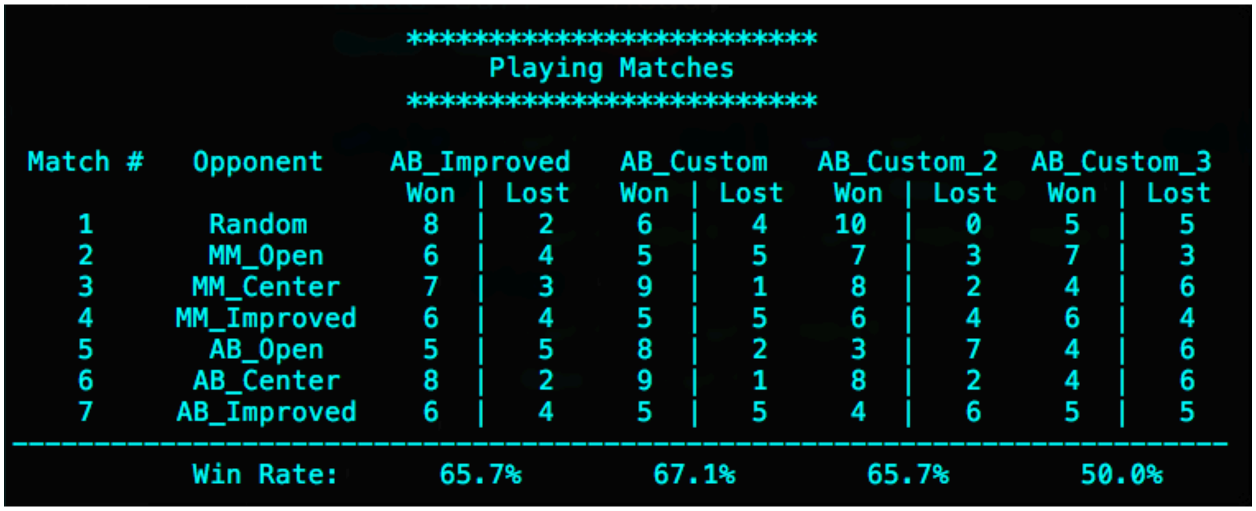
Fig. 1. Tournament 1.



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

 Match #     Opponent      AB_Improved    AB_Custom    AB_Custom_2   AB_Custom_3
                           Won | Lost    Won | Lost    Won | Lost    Won | Lost
    1         Random        8  |   2      7  |   3      9  |   1     10  |   0
    2         MM_Open       6  |   4      7  |   3      7  |   3      7  |   3
    3        MM_Center      8  |   2      9  |   1      8  |   2      7  |   3
    4       MM_Improved     6  |   4      5  |   5      6  |   4      6  |   4
    5         AB_Open       5  |   5      4  |   6      4  |   6      7  |   3
    6        AB_Center      7  |   3      7  |   3      6  |   4      5  |   5
    7       AB_Improved     4  |   6      5  |   5      6  |   4      7  |   3
--------------------------------------------------------------------------------
           Win Rate:         62.9%         62.9%         65.7%         70.0%
```

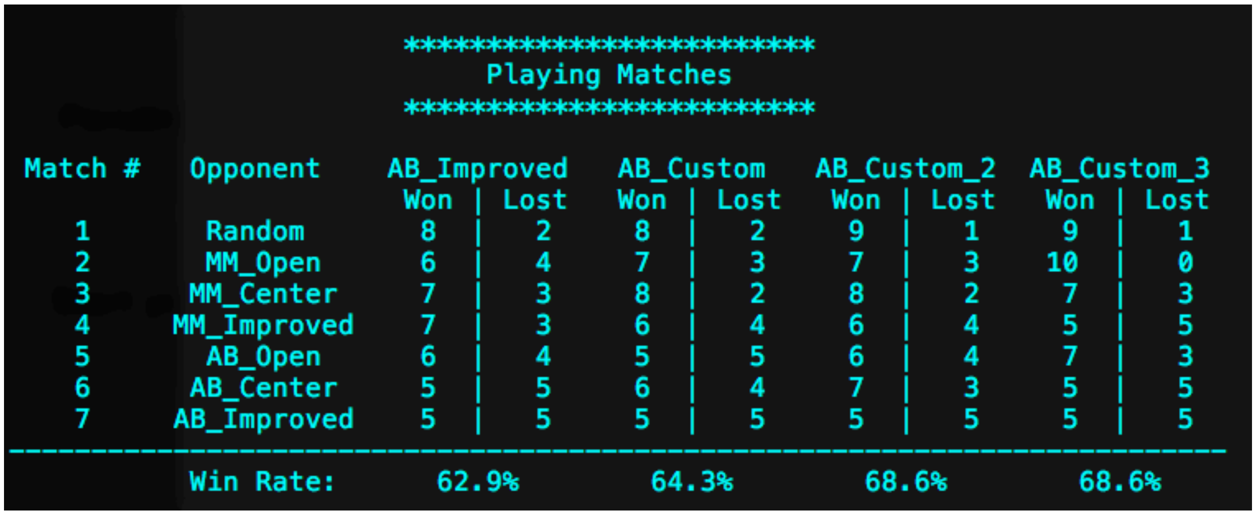
Fig. 2. Tournament 2.



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

 Match #     Opponent     AB_Improved   AB_Custom   AB_Custom_2  AB_Custom_3
                           Won | Lost   Won | Lost   Won | Lost   Won | Lost
    1         Random        8  |  2      6  |  4     10  |  0      5  |  5
    2         MM_Open       6  |  4      5  |  5      7  |  3      7  |  3
    3        MM_Center      7  |  3      9  |  1      8  |  2      4  |  6
    4       MM_Improved     6  |  4      5  |  5      6  |  4      6  |  4
    5         AB_Open       5  |  5      8  |  2      3  |  7      4  |  6
    6        AB_Center      8  |  2      9  |  1      8  |  2      4  |  6
    7       AB_Improved     6  |  4      5  |  5      4  |  6      5  |  5
--------------------------------------------------------------------------
           Win Rate:      65.7%        67.1%        65.7%        50.0%
```

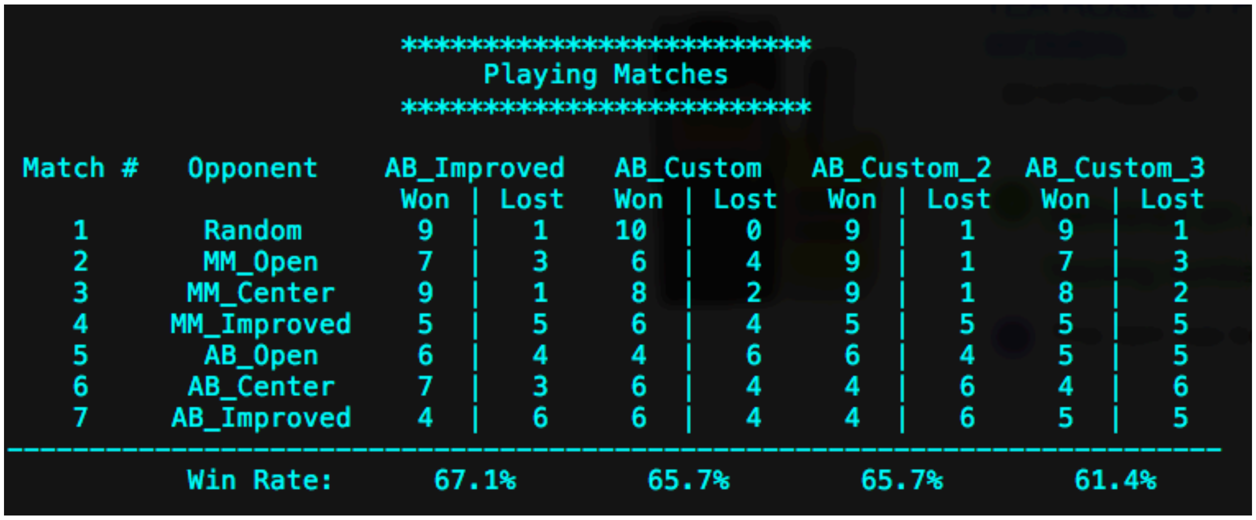
Fig. 3. Tournament 3.



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

 Match #     Opponent     AB_Improved   AB_Custom   AB_Custom_2  AB_Custom_3
                           Won | Lost   Won | Lost   Won | Lost   Won | Lost
    1         Random        8  |  2      8  |  2      9  |  1      9  |  1
    2         MM_Open       6  |  4      7  |  3      7  |  3     10  |  0
    3        MM_Center      7  |  3      8  |  2      8  |  2      7  |  3
    4       MM_Improved     7  |  3      6  |  4      6  |  4      5  |  5
    5         AB_Open       6  |  4      5  |  5      6  |  4      7  |  3
    6        AB_Center      5  |  5      6  |  4      7  |  3      5  |  5
    7       AB_Improved     5  |  5      5  |  5      5  |  5      5  |  5
--------------------------------------------------------------------------
           Win Rate:      62.9%        64.3%        68.6%        68.6%
```

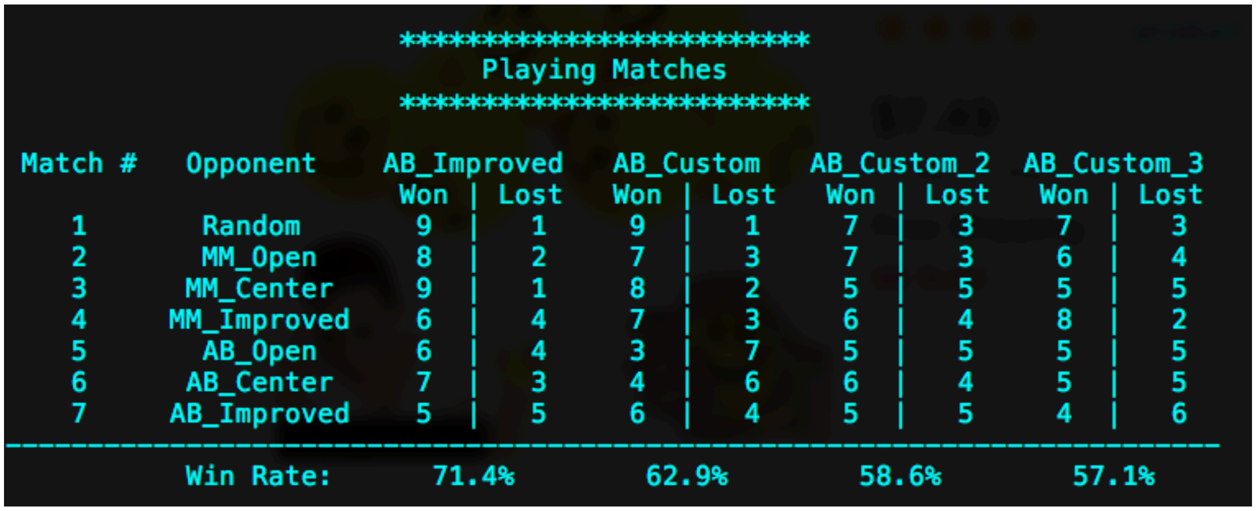
Fig. 4. Tournament 4.



```
                         **************************
                              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     10  |  0      9  |  1      9  |  1
    2         MM_Open       7  |  3      6  |  4      9  |  1      7  |  3
    3        MM_Center      9  |  1      8  |  2      9  |  1      8  |  2
    4       MM_Improved     5  |  5      6  |  4      5  |  5      5  |  5
    5         AB_Open       6  |  4      4  |  6      6  |  4      5  |  5
    6        AB_Center      7  |  3      6  |  4      4  |  6      4  |  6
    7       AB_Improved     4  |  6      6  |  4      4  |  6      5  |  5
--------------------------------------------------------------------------
           Win Rate:      67.1%        65.7%        65.7%        61.4%
```

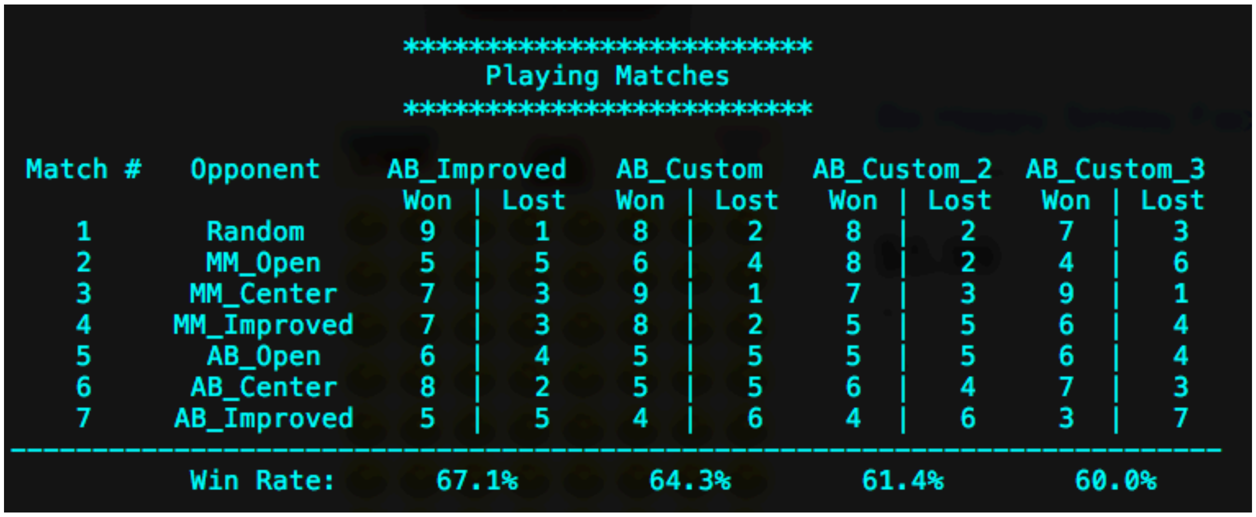
Fig. 5. Tournament 5.



```
                          *************************
                               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      7  |  3      7  |  3
    2        MM_Open      8  |  2      7  |  3      7  |  3      6  |  4
    3       MM_Center     9  |  1      8  |  2      5  |  5      5  |  5
    4      MM_Improved    6  |  4      7  |  3      6  |  4      8  |  2
    5        AB_Open      6  |  4      3  |  7      5  |  5      5  |  5
    6       AB_Center     7  |  3      4  |  6      6  |  4      5  |  5
    7      AB_Improved    5  |  5      6  |  4      5  |  5      4  |  6
--------------------------------------------------------------------------
           Win Rate:      71.4%        62.9%        58.6%        57.1%
```

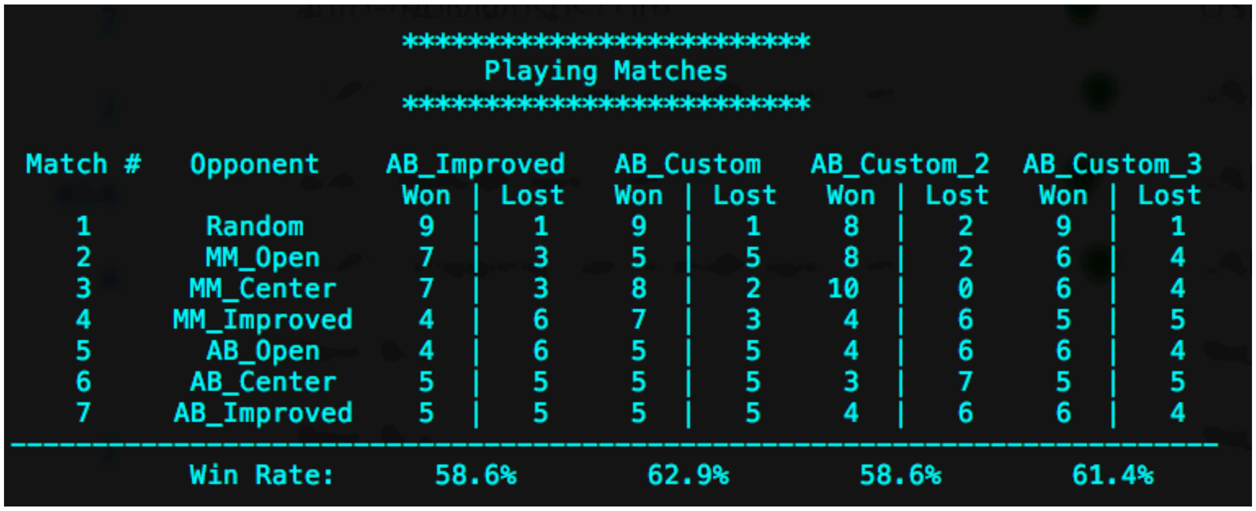
Fig. 6. Tournament 6.



```
                          *************************
                               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      8  |  2      8  |  2      7  |  3
    2        MM_Open      5  |  5      6  |  4      8  |  2      4  |  6
    3       MM_Center     7  |  3      9  |  1      7  |  3      9  |  1
    4      MM_Improved    7  |  3      8  |  2      5  |  5      6  |  4
    5        AB_Open      6  |  4      5  |  5      5  |  5      6  |  4
    6       AB_Center     8  |  2      5  |  5      6  |  4      7  |  3
    7      AB_Improved    5  |  5      4  |  6      4  |  6      3  |  7
--------------------------------------------------------------------------
           Win Rate:      67.1%        64.3%        61.4%        60.0%
```

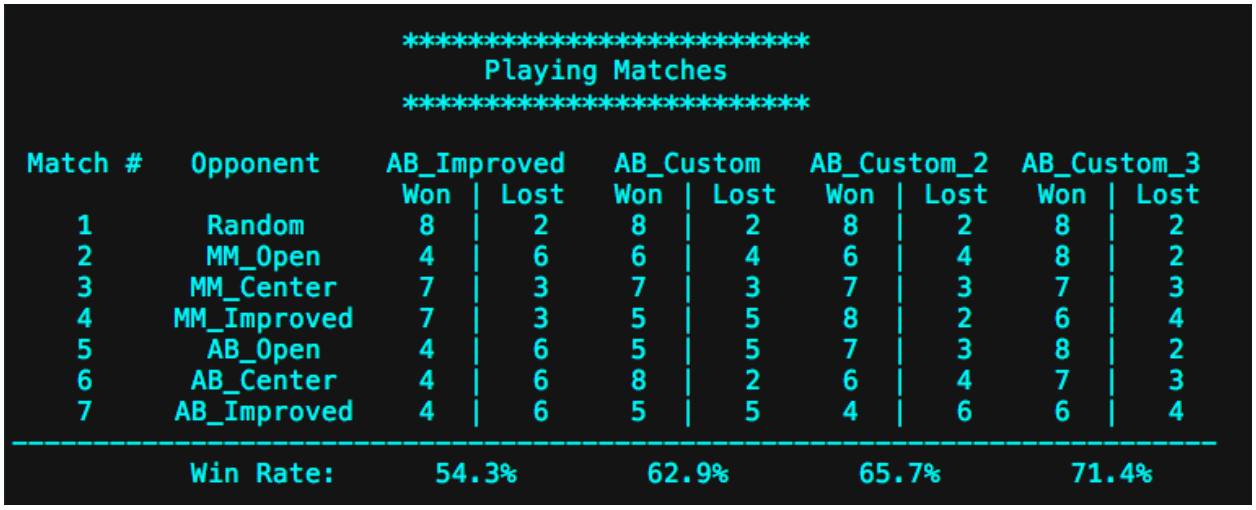
Fig. 7. Tournament 7.



```
                          *************************
                               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      5  |  5      8  |  2      6  |  4
    3       MM_Center     7  |  3      8  |  2     10  |  0      6  |  4
    4      MM_Improved    4  |  6      7  |  3      4  |  6      5  |  5
    5        AB_Open      4  |  6      5  |  5      4  |  6      6  |  4
    6       AB_Center     5  |  5      5  |  5      3  |  7      5  |  5
    7      AB_Improved    5  |  5      5  |  5      4  |  6      6  |  4
--------------------------------------------------------------------------
           Win Rate:      58.6%        62.9%        58.6%        61.4%
```

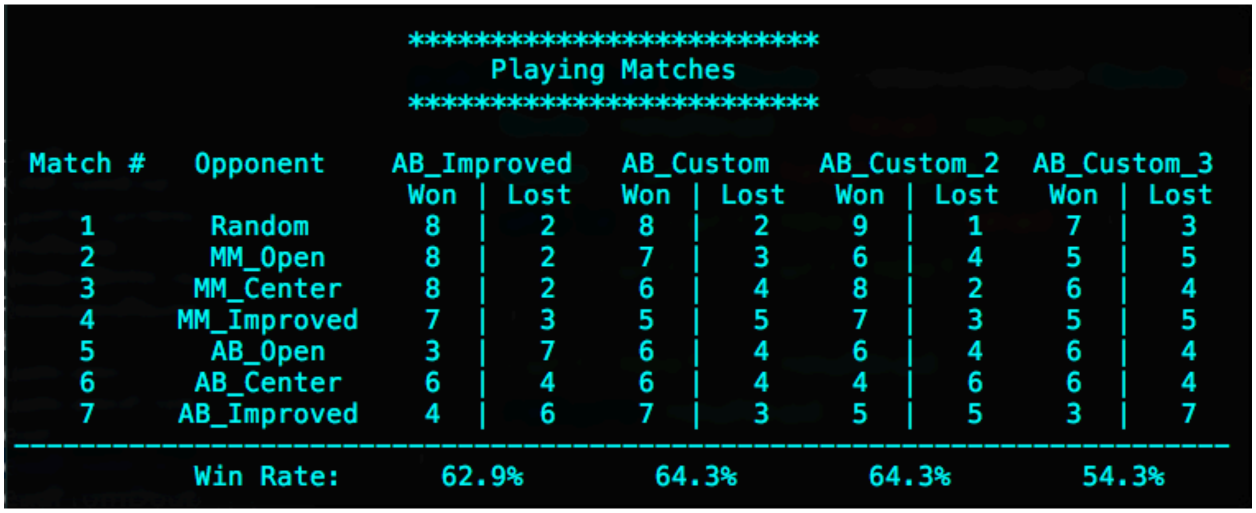
Fig. 8. Tournament 8.



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

 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      8  |  2      8  |  2
    2        MM_Open        4  |  6      6  |  4      6  |  4      8  |  2
    3       MM_Center       7  |  3      7  |  3      7  |  3      7  |  3
    4      MM_Improved      7  |  3      5  |  5      8  |  2      6  |  4
    5        AB_Open        4  |  6      5  |  5      7  |  3      8  |  2
    6       AB_Center       4  |  6      8  |  2      6  |  4      7  |  3
    7      AB_Improved      4  |  6      5  |  5      4  |  6      6  |  4
--------------------------------------------------------------------------
           Win Rate:      54.3%        62.9%        65.7%        71.4%
```

Fig. 9. Tournament 9.



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

 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      7  |  3
    2        MM_Open        8  |  2      7  |  3      6  |  4      5  |  5
    3       MM_Center       8  |  2      6  |  4      8  |  2      6  |  4
    4      MM_Improved      7  |  3      5  |  5      7  |  3      5  |  5
    5        AB_Open        3  |  7      6  |  4      6  |  4      6  |  4
    6       AB_Center       6  |  4      6  |  4      4  |  6      6  |  4
    7      AB_Improved      4  |  6      7  |  3      5  |  5      3  |  7
--------------------------------------------------------------------------
           Win Rate:      62.9%        64.3%        64.3%        54.3%
```

Fig. 10. Tournament 10.

As shown in Figs. 1–10 above and summarized in Table 1 below, the performance of the game agents using `improved_score()` function and three custom score functions varied widely between each tournament. In particular, the win rates for `AB_Improved` base agent varied between 54.3% and 71.4%. Similarly, the win rates for `AB_Custom_3` agent varied between 50.0% and 71.4%. There was a 10% variation in the win rates for `AB_Custom_2` agent, ranging from 58.6% to 68.6%. In contrast, `AB_Custom` agent showed the most consistent performance across tournaments, with the smallest difference between the lowest win rate (62.9%) and the highest win rate (67.1%). Also interesting is the fact that the lowest win rate of 62.9% for the `AB_Custom` agent resulted in 5 out of 10 tournaments, which seems to suggest the possibility that it may be expected for the agent to win at least 62.9% of the games in most of the times.

Table 1. Win rates per tournament (Highest win rates for each game agent across tournaments are highlighted in green; lowest win rates for each game agent are highlighted in red.)

Tournament	AB_Improved	AB_Custom	AB_Custom_2	AB_Custom_3
1	65.7%	62.9%	62.9%	58.6%
2	62.9%	62.9%	65.7%	70.0%
3	65.7%	67.1%	65.7%	50.0%
4	62.9%	64.3%	68.6%	68.6%
5	67.1%	65.7%	65.7%	61.4%
6	71.4%	62.9%	58.6%	57.1%
7	67.1%	64.3%	61.4%	60.0%
8	58.6%	62.9%	58.6%	61.4%
9	54.3%	62.9%	65.7%	71.4%
10	62.9%	64.3%	64.3%	54.3%

One common factor among the test agents was that overall they competed best against the `Random` agent, followed by `MM` agents, and worst against the `AB` agents, which was most clearly noticeable in the case of `AB_Custom_2`, as shown in Table 2 and Fig. 11. In terms of score functions (excluding random), the test agents tended to be strongest against the agents using `open_score()` function for both `MM` and `AB` opponents. This may suggest that the four score functions used by the test agents, all of which are based on the consideration of the number of moves, in some form, may be competitive against the distance-based `open_score()` function.

Table 2. Wins per opponent

Opponent	AB_Improved	AB_Custom	AB_Custom_2	AB_Custom_3
Random	84	81	85	77
MM_Open	62	61	72	66
MM_Center	78	80	79	66
MM_Improved	62	62	58	60
AB_Open	51	50	52	57
AB_Center	63	63	54	54
AB_Improved	47	51	46	49

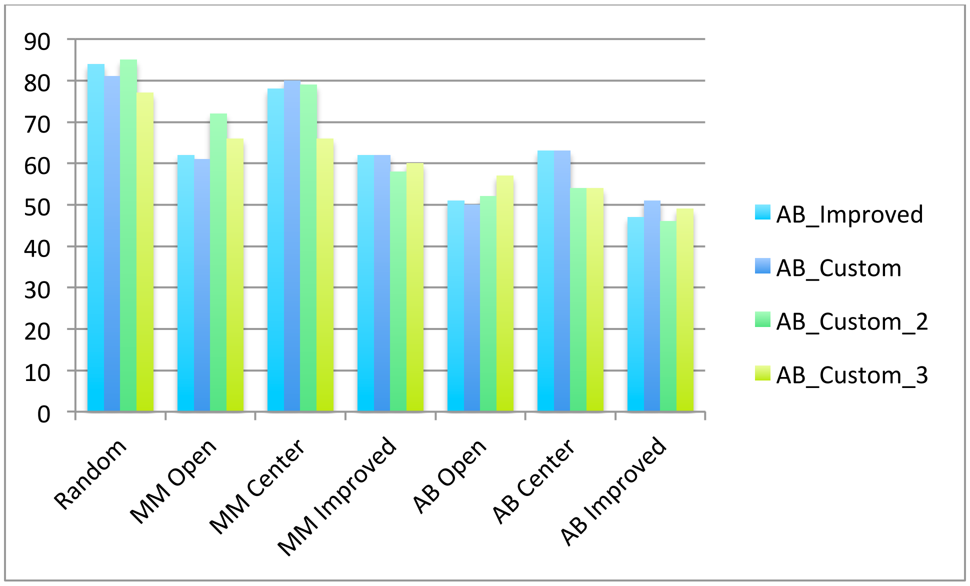


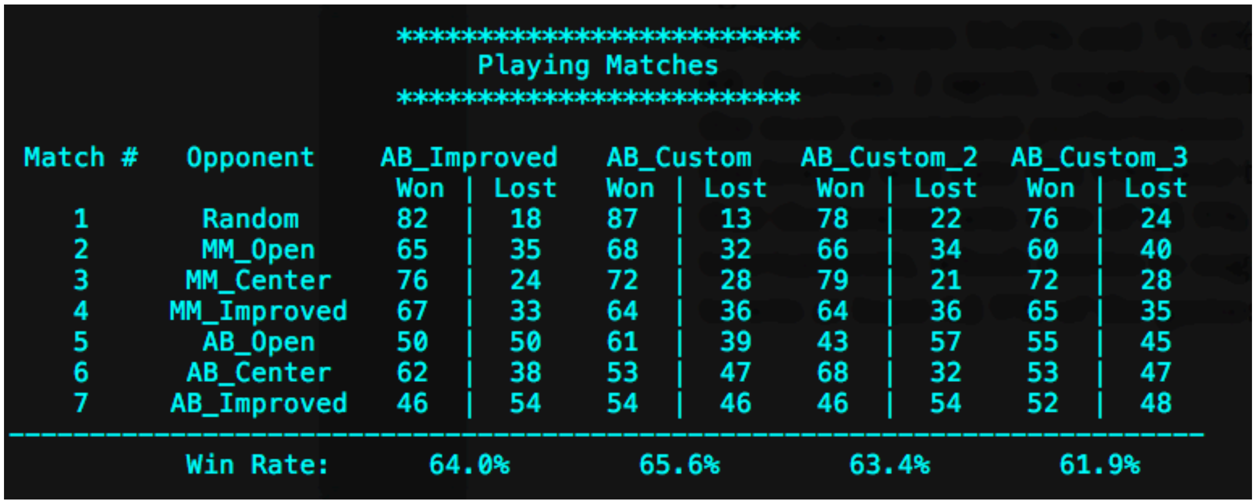
Fig. 11. Aggregated performance of test agents per opponent.

Table 3 summarizes the number of wins and average win rates per test agent, which shows very close performance results for `AB_Improved`, `AB_Custom`, and `AB_Custom_2`.

Table 3. Total wins and win rates

AB_Improved	AB_Custom	AB_Custom_2	AB_Custom_3
447	448	446	429
63.9%	64.0%	63.7%	61.3%

Figs. 12 and 13 present the results of the final tournament with the increased size of 100 games per opponent, which show that `AB_Custom` agent performed more noticeably better than `AB_Improved` agent.


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

 Match #     Opponent      AB_Improved    AB_Custom     AB_Custom_2   AB_Custom_3
                            Won | Lost    Won | Lost    Won | Lost    Won | Lost
    1         Random        82  |  18     87  |  13     78  |  22     76  |  24
    2         MM_Open       65  |  35     68  |  32     66  |  34     60  |  40
    3        MM_Center      76  |  24     72  |  28     79  |  21     72  |  28
    4       MM_Improved     67  |  33     64  |  36     64  |  36     65  |  35
    5         AB_Open       50  |  50     61  |  39     43  |  57     55  |  45
    6        AB_Center      62  |  38     53  |  47     68  |  32     53  |  47
    7       AB_Improved     46  |  54     54  |  46     46  |  54     52  |  48
--------------------------------------------------------------------------
           Win Rate:          64.0%         65.6%         63.4%         61.9%
```

Fig. 12. Tournament with 100 games per opponent.

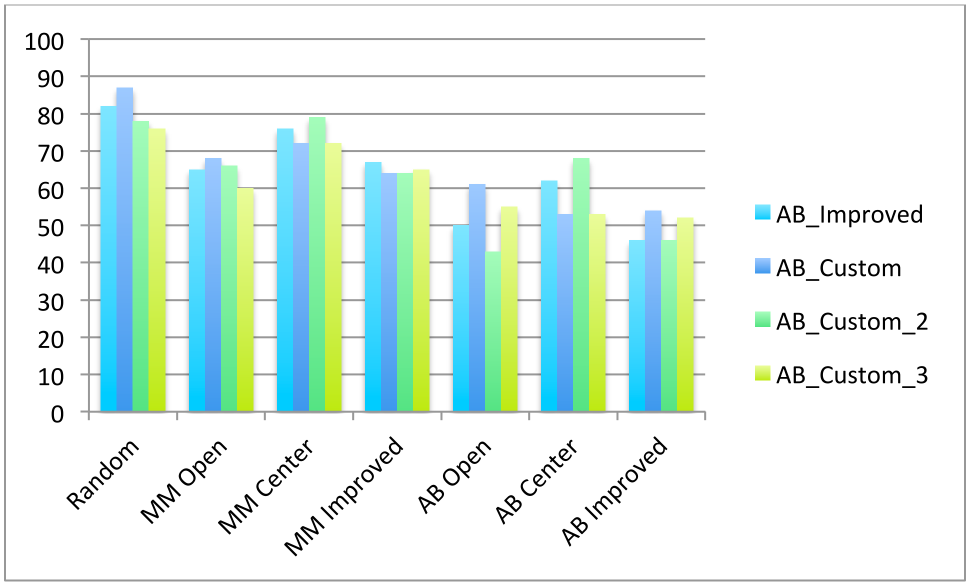


Fig. 13. Performance of test agents per opponent in the final tournament.

3. Discussion

Based on the experimental results, it is recommended that `custom_score()` function be used as the heuristic evaluation function. The reasons are: (1) `AB_Custom` agent performed better than `AB_Improved` agent, both in the aggregated results of smaller tournaments and in the results of the final larger tournament. (2) `AB_Custom` agent also showed most consistent performance across tournaments. (3) The `custom_score()` function used by `AB_Custom` agent provides a more fine-grained differentiation between game states that result in the same score when using `improved_score()`, which may explain the relatively superior performance by `AB_Custom` agent.

One thing to note about the three custom score functions, as well as `improved_score()`, is the fact that all these heuristics are based on the computation of the number of moves available at a given game state, without considering other factors. Even though `one_step_look_ahead_score()` also considers the favorability or unfavorability of the forecast game state, the determination is also based on the number of moves available in the game state.

Perhaps considering the game state from different perspectives, including heuristic determination of the overall “health” of the given state or predicted directionality of the game progress, may lead to an improved score function. In this regard, it may be mentioned that approaches that apply deep learning by using convolutional and recurrent neural networks were attempted initially. Those approaches, however, have been discarded without much further exploration, due to the limitations imposed on the modifiability and compatibility of code (i.e., only `game_agent.py` can be effectively modified and the resulting code should run on the grading system without the need for additional libraries.).