Heuristic analysis

As we know, the point of isolation is to illuminate the opponent's moves. A good evaluation function help us seek the most options while trying to get in the way of the opponent's moves. All these related to my moves and opponent moves. In this case, all my Heuristic based on the own_moves and opp_moves, here is how to compute:

```
if game.is_loser(player):
        return float("-inf")

if game.is_winner(player):
        return float("inf")

own_moves = len(game.get_legal_moves(player))
opp_moves = len(game.get_legal_moves(game.get_opponent(player)))
```

Based on the information, we can try the following heuristic:

Heuristic1: only consider my moves
custom_score:

```
float((own_moves+1)/(opp_moves+1) - 1)
```

Heuristic2: only consider opponent moves
custom_score_2:

```
return float(-opp_moves)
```

Heuristic3: consider both my moves and opponent moves
custom_score_3:

```
return float(3*own_moves - 2*opp_moves)
```

The output is:

```
                    ************************
                         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      7  |   3     8  |   2     3  |   7     7  |   3
    3      MM_Center     8  |   2     8  |   2     8  |   2     6  |   4
    4     MM_Improved    6  |   4     6  |   4     4  |   6     6  |   4
    5       AB_Open      5  |   5     4  |   6     7  |   3     5  |   5
    6      AB_Center     4  |   6     8  |   2     7  |   3     8  |   2
    7     AB_Improved    5  |   5     4  |   6     5  |   5     5  |   5
--------------------------------------------------------------------------
           Win Rate:      61.4%        65.7%        61.4%        62.9%
```

Heuristic1 use my moves/opponent moves, it would help us seek most options over the opponent's behavior. It works well in Radom, MM_Open, MM_Center and AB_Center, work badly in the others.

Heuristic2 consider only opponent moves, it would chase after the opponent's behavior. It works well in Random and MM_Center, work badly in the others.

Heuristic3 conside both my moves and opponent moves, it would balance the tradeoff between seek most options and opponent's behavior. It works well in AB_Center.

All these heuristics are easy to implement and avoid timeout in other algorithm. Especially, we can find Heuristic1 had higher score than Heuristic2 and Heuristic3. It also help agent win the most opponent among all the algorithm. Thus, I would recommend Heuristic1 as the evaluation function.