Heuristic analysis

Dexiang Xu

1.Synopsis

The project is aiming to develop an adversarial search agent to play the game "Isolation".

Isolation is a deterministic, two-player game of perfect information in which the players alternate turns moving a single piece from one cell to another on a board. Whenever either player occupies a cell, that cell becomes blocked for the remainder of the game. The first player with no remaining legal moves loses, and the opponent is declared the winner. These rules are implemented in the “isolation.Board” class provided in the repository.

This project uses a version of Isolation where each agent is restricted to L-shaped movements (like a knight in chess) on a rectangular grid (like a chess or checkerboard). The agents can move to any open cell on the board that is 2-rows and 1-column or 2-columns and 1-row away from their current position on the board. Movements are blocked at the edges of the board (the board does not wrap around), however, the player can "jump" blocked or occupied spaces (just like a knight in chess).

Additionally, agents will have a fixed time limit each turn to search for the best move and respond. If the time limit expires during a player's turn, that player forfeits the match, and the opponent wins.

2. Heuristic

Customer player1:

This heuristic is trying to maximizing player’s move, the expression is as below:

my_available_moves - 1.5 * opponent_available_moves

Customer player2:

This heuristic is trying to make potential moves to be close to the center of the board as possible:

my_available_moves - opponent_available_moves + center_score()

Customer player3:

This heuristic can be expressed as:

$$1.5 * \frac{\text{My\_available\_moves}}{\text{Opponent\_available\_moves}} - \frac{\text{Opponent\_available\_moves}}{\text{My\_available\_moves}}$$

as, trying to maximizing the ration of player's move compare to opponent's, and minimizing the ration of opponent's move compare to player's move.

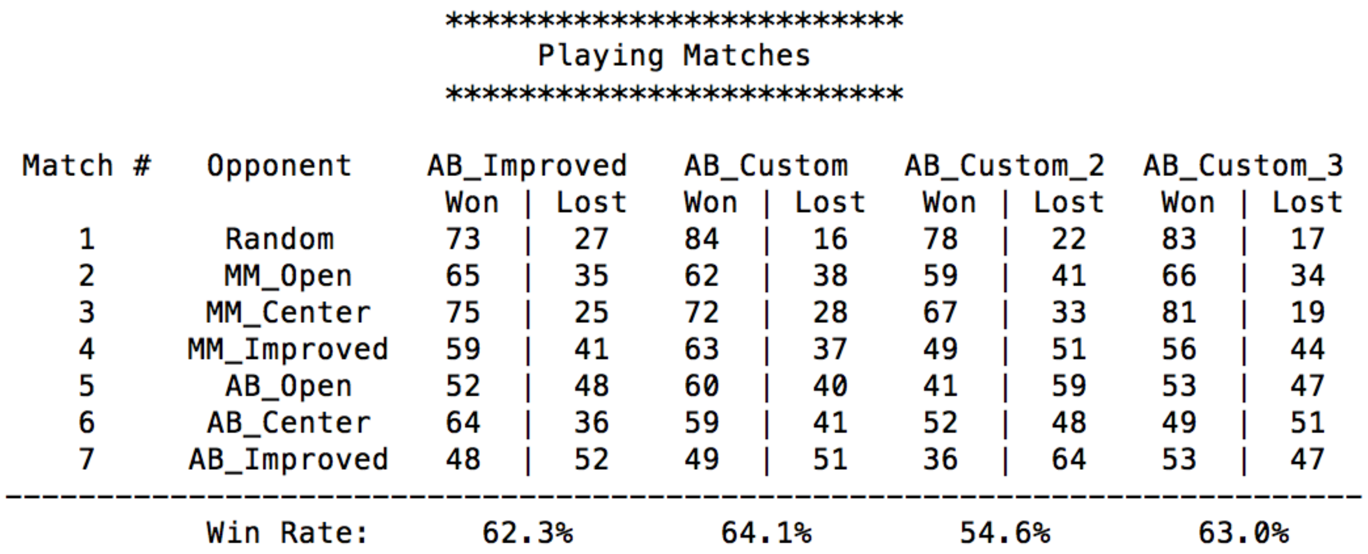
3. Result

The relative performance of the game agent is tested in a round-robin tournament against several other pre-defined agents. And the opponents are listed below:

1. Random: An agent that randomly chooses a move each turn.
2. MM_Open: MinimaxPlayer agent using the open_move_score heuristic with search depth 3
3. MM_Center: MinimaxPlayer agent using the center_score heuristic with search depth 3
4. MM_Improved: MinimaxPlayer agent using the improved_score heuristic with search depth 3
5. AB_Open: AlphaBetaPlayer using iterative deepening alpha-beta search and the open_move_score heuristic
6. AB_Center: AlphaBetaPlayer using iterative deepening alpha-beta search and the center_score heuristic
7. AB_Improved: AlphaBetaPlayer using iterative deepening alpha-beta search and the improved_score heuristic

And to get a better performance and larger sample size, the number of matches is set to 50, and time limit for each search was set to 200 ms.

The result has been listed below:



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

 Match #   Opponent    AB_Improved   AB_Custom   AB_Custom_2  AB_Custom_3
                        Won | Lost   Won | Lost   Won | Lost   Won | Lost
    1       Random      73  |  27    84  |  16    78  |  22    83  |  17
    2       MM_Open     65  |  35    62  |  38    59  |  41    66  |  34
    3      MM_Center    75  |  25    72  |  28    67  |  33    81  |  19
    4     MM_Improved   59  |  41    63  |  37    49  |  51    56  |  44
    5       AB_Open     52  |  48    60  |  40    41  |  59    53  |  47
    6      AB_Center    64  |  36    59  |  41    52  |  48    49  |  51
    7     AB_Improved   48  |  52    49  |  51    36  |  64    53  |  47
--------------------------------------------------------------------------
           Win Rate:      62.3%        64.1%        54.6%        63.0%
```

Overall, the third heuristic has the best performance. and AB_center and AB_Improved has similar performance against the three custom players. And custom player generally have better performance against Minmax-player. But overall, they have very similar win rate, except the custom player two.