28th December 2017

# Isolation Heuristic Review

By: Atri Basu, Udacity AI Nandodegree, Term 1

Learning from the heuristics:

1. Each strategy performs consistently well against certain types of counter strategies. Therefor having a way to pick your heuristic based on a best guess as to kind of strategy being used by your opponent could greatly enhance the chances of the computer player winning
2. Making the computer player aggressively(multipliers) pursue a certain heuristic does tend to make the computer more competitive, but beyond a certain level aggression doesn't necessarily add value.
3. Aggressive heuristics also led to reduced forfeitures.
4. However playing defensively(dividing own heuristic) didn't improve the machines win rate and led to more forfeitures than when playing aggressively
5. Based on the multiple tests carried out I would recommend the difference between the number of legal moves and twice that of the opponent because:
    a. It consistently posted the highest win rate
    b. It was stable. The machine remained undefeated across multiple iterations and avatars of this heuristic.
    c. It was computationally simple

Kinds of heuristics Tested:

- Losing match ups are marked in red
- Undefeated Win rates are marked in green
- Lowest Forfeitures is underscored in yellow

1. Number of moves:
    a. Number of blank spaces
    b. Number of legal moves
    c. Distance from center

```
(aind) C:\Users\atbasu\Documents\AI\Udacity\AIND-Isolation>tournament.py

This script evaluates the performance of the custom_score evaluation
function against a baseline agent using alpha-beta search and iterative
deepening (ID) called `AB_Improved`. The three `AB_Custom` agents use
ID and alpha-beta search with the custom_score functions defined in
game_agent.py.
[O]LM → [Opponent's] Legal Moves
[O]BS → [Opponent's] Blank Spaces
[O]Dc → [Opponent's]  Distance from Center


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

```
 Match #    Opponent     AB_Improved       BS            LM            Dc
                          Won | Lost    Won | Lost    Won | Lost    Won | Lost
    1       Random         8  |   2      6  |   4      8  |   2      7  |   3
    2       MM_Open        7  |   3      7  |   3      6  |   4      5  |   5
    3      MM_Center       7  |   3      7  |   3      8  |   2      6  |   4
    4     MM_Improved      6  |   4      6  |   4      8  |   2      5  |   5
    5       AB_Open        4  |   6      5  |   5      5  |   5      4  |   6
    6      AB_Center       5  |   5      5  |   5      8  |   2      7  |   3
    7     AB_Improved      4  |   6      5  |   5      6  |   4      4  |   6
--------------------------------------------------------------------------------
           Win Rate:      58.6%        58.6%        70.0%        54.3%

Your agents forfeited 173.0 games while there were still legal moves available to play.
```

2. Difference between number of moves
    a. Number of legal moves
    b. Distance from Center
    c. Blank spaces and opponents legal moves

```
(aind) C:\Users\atbasu\Documents\AI\Udacity\AIND-Isolation>tournament.py

This script evaluates the performance of the custom_score evaluation
function against a baseline agent using alpha-beta search and iterative
deepening (ID) called `AB_Improved`. The three `AB_Custom` agents use
ID and alpha-beta search with the custom_score functions defined in
game_agent.py.
[O]LM → [Opponent's] Legal Moves
[O]BS → [Opponent's] Blank Spaces
[O]Dc → [Opponent's]  Distance from Center

                ********************************************
                   Paying Attention To Opponent in Matches
                ********************************************

 Match #    Opponent     AB_Improved     LM - OLM      Dc - ODc       BS - OLM
                          Won | Lost    Won | Lost    Won | Lost     Won | Lost
    1       Random         7  |   3      8  |   2      8  |   2       5  |   5
    2       MM_Open        5  |   5      6  |   4      4  |   6       5  |   5
    3      MM_Center       9  |   1      7  |   3      9  |   1       6  |   4
    4     MM_Improved      4  |   6      3  |   7      7  |   3       5  |   5
    5       AB_Open        6  |   4      5  |   5      5  |   5       6  |   4
    6      AB_Center       6  |   4      5  |   5      4  |   6       4  |   6
    7     AB_Improved      5  |   5      6  |   4      5  |   5       4  |   6
--------------------------------------------------------------------------------
           Win Rate:      60.0%        57.1%        60.0%          50.0%

Your agents forfeited 182.0 games while there were still legal moves available to play.
```

3. Combo Heuristics
    a. Fraction of blanks spaces that are legal moves
    b. Number of legal moves divided by the distance from center
    c. Difference between players Fraction of blanks spaces that are legal moves

```
(aind) C:\Users\atbasu\Documents\AI\Udacity\AIND-Isolation>tournament.py

This script evaluates the performance of the custom_score evaluation
function against a baseline agent using alpha-beta search and iterative
deepening (ID) called `AB_Improved`. The three `AB_Custom` agents use
ID and alpha-beta search with the custom_score functions defined in
game_agent.py.
```

```
[O]LM → [Opponent's] Legal Moves
[O]BS → [Opponent's] Blank Spaces
[O]Dc → [Opponent's]  Distance from Center

                    **************************
                          Combos
                    **************************

 Match #    Opponent    AB_Improved     LM/BS      LM/Dc - OLM   LM/BS - OLM
                         Won | Lost    Won | Lost   Won | Lost   Won | Lost
    1       Random        8  |   2     9  |   1     8  |   2     10 |   0
    2       MM_Open       5  |   5     4  |   6     6  |   4     7  |   3
    3      MM_Center      8  |   2     6  |   4     4  |   6     8  |   2
    4     MM_Improved     7  |   3     6  |   4     6  |   4     5  |   5
    5       AB_Open       2  |   8     5  |   5     5  |   5     7  |   3
    6      AB_Center      6  |   4     5  |   5     8  |   2     6  |   4
    7     AB_Improved     6  |   4     4  |   6     5  |   5     4  |   6
-------------------------------------------------------------------------
           Win Rate:      60.0%        55.7%        60.0%        67.1%

Your agents forfeited 168.0 games while there were still legal moves available to play.
```

## 4. Aggression

### a. Aggressive Combo Heurists:

```
(aind) C:\Users\atbasu\Documents\AI\Udacity\AIND-Isolation>tournament.py

This script evaluates the performance of the custom_score evaluation
function against a baseline agent using alpha-beta search and iterative
deepening (ID) called `AB_Improved`. The three `AB_Custom` agents use
ID and alpha-beta search with the custom_score functions defined in
game_agent.py.
[O]LM → [Opponent's] Legal Moves
[O]BS → [Opponent's] Blank Spaces
[O]Dc → [Opponent's]  Distance from Center

                    **************************
                         Attacking Combos
                    **************************

 Match #    Opponent    AB_Improved   LM - 2*OLM   LM/Dc -2*OLM LM/BS - 2*OLM
                         Won | Lost    Won | Lost   Won | Lost   Won | Lost
    1       Random        9  |   1     8  |   2     9  |   1     8  |   2
    2       MM_Open       5  |   5     6  |   4     6  |   4     6  |   4
    3      MM_Center      7  |   3     9  |   1     8  |   2     8  |   2
    4     MM_Improved     7  |   3     8  |   2     5  |   5     7  |   3
    5       AB_Open       6  |   4     6  |   4     5  |   5     5  |   5
    6      AB_Center      7  |   3     7  |   3     7  |   3     6  |   4
    7     AB_Improved     7  |   3     5  |   5     6  |   4     6  |   4
-------------------------------------------------------------------------
           Win Rate:      68.6%        70.0%        65.7%        65.7%

Your agents forfeited 165.0 games while there were still legal moves available to
play.
```

### b. Aggressive OLM:

```
(aind) C:\Users\atbasu\Documents\AI\Udacity\AIND-Isolation>tournament.py

This script evaluates the performance of the custom_score evaluation
function against a baseline agent using alpha-beta search and iterative
deepening (ID) called `AB_Improved`. The three `AB_Custom` agents use
ID and alpha-beta search with the custom_score functions defined in
```

game_agent.py.
[O]LM → [Opponent's] Legal Moves

**********
Attacking Matches
**********

| Match # | Opponent | AB_Improved | | LM - 2*OLM | | LM - 4*OLM | | LM - 8*OLM | |
|---|---|---|---|---|---|---|---|---|---|
| | | Won | Lost | Won | Lost | Won | Lost | Won | Lost |
| 1 | Random | 7 | 3 | 10 | 0 | 8 | 2 | 8 | 2 |
| 2 | MM_Open | 6 | 4 | 6 | 4 | 7 | 3 | 8 | 2 |
| 3 | MM_Center | 8 | 2 | 7 | 3 | 8 | 2 | 7 | 3 |
| 4 | MM_Improved | 7 | 3 | 6 | 4 | 8 | 2 | 4 | 6 |
| 5 | AB_Open | 4 | 6 | 8 | 2 | 7 | 3 | 4 | 6 |
| 6 | AB_Center | 6 | 4 | 6 | 4 | 6 | 4 | 8 | 2 |
| 7 | AB_Improved | 3 | 7 | 6 | 4 | 5 | 5 | 6 | 4 |
| | Win Rate: | 58.6% | | 70.0% | | 70.0% | | 64.3% | |

Your agents forfeited 165.0 games while there were still legal moves available to play.

**********
Defensive Matches
**********

| Match # | Opponent | AB_Improved | | LM/2 - OLM | | LM/2 - 2*OLM | | LM/4-OLM | |
|---|---|---|---|---|---|---|---|---|---|
| | | Won | Lost | Won | Lost | Won | Lost | Won | Lost |
| 1 | Random | 8 | 2 | 10 | 0 | 6 | 4 | 7 | 3 |
| 2 | MM_Open | 5 | 5 | 6 | 4 | 6 | 4 | 7 | 3 |
| 3 | MM_Center | 7 | 3 | 8 | 2 | 8 | 2 | 6 | 4 |
| 4 | MM_Improved | 5 | 5 | 5 | 5 | 6 | 4 | 6 | 4 |
| 5 | AB_Open | 5 | 5 | 6 | 4 | 6 | 4 | 5 | 5 |
| 6 | AB_Center | 8 | 2 | 4 | 6 | 5 | 5 | 5 | 5 |
| 7 | AB_Improved | 2 | 8 | 5 | 5 | 5 | 5 | 7 | 3 |
| | Win Rate: | 57.1% | | 62.9% | | 60.0% | | 61.4% | |

Your agents forfeited 174.0 games while there were still legal moves available to play.