

Comparison of heuristic methods applied to Isolation

April 15, 2017

1 Methods

1.1 Number available moves

The simplest of methods available is the number of moves available to the player in the current state. This is a simple metric and does not capture a complete state of the board, not taking into account the relative state of the opponent.

1.2 Area based heuristic

This is a more complex method, working out the number of empty positions around the current player, the downside of this method is that it is relatively expensive.

1.3 Axis of movement based heuristic

Similar to the number of available moves this method attempts to get more granular detail of the game state by taking into account the directions of movement available to the player. Two methods were implemented for this the first returning the product of the movement along each direction the second from the formula:

$$\frac{\text{num_axis_available}}{8} \times \text{num_moves_available} \quad (1)$$

1.4 Adversarial implementation

The above heuristics do not take into account adequately the opponent state. When minimax takes this into account this is relatively more expensive as the bulk of computational expense is in the last layer of the explored space. The solution is to lazily factor in the opponent by applying the same heuristic to the opponent, multiplied by an arbitrary priority coefficient (this is to emphasize/deemphasize the opponents state on the board), as if it were the opponent turn.

2 Results (adversarial results only)

NB: all coefficient multiplier for all methods was 2

2.1 Number of available moves

```
*****
Evaluating: ID_Improved
*****
```

Playing Matches:

```
-----
Match 1: ID_Improved vs Random      Result: 18 to 2
Match 2: ID_Improved vs MM_Null     Result: 18 to 2
Match 3: ID_Improved vs MM_Open     Result: 15 to 5
Match 4: ID_Improved vs MM_Improved Result: 14 to 6
Match 5: ID_Improved vs AB_Null     Result: 18 to 2
Match 6: ID_Improved vs AB_Open     Result: 13 to 7
Match 7: ID_Improved vs AB_Improved Result: 11 to 9
```

Results:

```
-----
ID_Improved          76.43%
```

```
*****
Evaluating: Student
*****
```

Playing Matches:

```
-----
Match 1: Student vs Random      Result: 17 to 3
Match 2: Student vs MM_Null     Result: 18 to 2
Match 3: Student vs MM_Open     Result: 10 to 10
Match 4: Student vs MM_Improved Result: 16 to 4
Match 5: Student vs AB_Null     Result: 18 to 2
Match 6: Student vs AB_Open     Result: 14 to 6
Match 7: Student vs AB_Improved Result: 17 to 3
```

Results:

```
-----
Student              78.57%
```

2.2 Area based heuristic

```
*****
Evaluating: ID_Improved
*****
```

Playing Matches:

Match 1:	ID_Improved	vs	Random	Result: 17 to 3
Match 2:	ID_Improved	vs	MM_Null	Result: 20 to 0
Match 3:	ID_Improved	vs	MM_Open	Result: 13 to 7
Match 4:	ID_Improved	vs	MM_Improved	Result: 15 to 5
Match 5:	ID_Improved	vs	AB_Null	Result: 18 to 2
Match 6:	ID_Improved	vs	AB_Open	Result: 12 to 8
Match 7:	ID_Improved	vs	AB_Improved	Result: 12 to 8

Results:

ID_Improved 76.43%

```
*****
Evaluating: Student
*****
```

Playing Matches:

Match 1:	Student	vs	Random	Result: 17 to 3
Match 2:	Student	vs	MM_Null	Result: 10 to 10
Match 3:	Student	vs	MM_Open	Result: 3 to 17
Match 4:	Student	vs	MM_Improved	Result: 2 to 18
Match 5:	Student	vs	AB_Null	Result: 9 to 11
Match 6:	Student	vs	AB_Open	Result: 4 to 16
Match 7:	Student	vs	AB_Improved	Result: 3 to 17

Results:

Student 34.29%

2.3 Axis of movement (using eqn 1)

```
*****
Evaluating: ID_Improved
*****
```

Playing Matches:

```
-----
Match 1: ID_Improved vs Random      Result: 19 to 1
Match 2: ID_Improved vs MM_Null     Result: 19 to 1
Match 3: ID_Improved vs MM_Open     Result: 12 to 8
Match 4: ID_Improved vs MM_Improved Result: 12 to 8
Match 5: ID_Improved vs AB_Null     Result: 18 to 2
Match 6: ID_Improved vs AB_Open     Result: 13 to 7
Match 7: ID_Improved vs AB_Improved Result: 13 to 7
```

Results:

```
-----
ID_Improved      75.71%
```

```
*****
Evaluating: Student
*****
```

Playing Matches:

```
-----
Match 1: Student vs Random      Result: 12 to 8
Match 2: Student vs MM_Null     Result: 11 to 9
Match 3: Student vs MM_Open     Result: 4 to 16
Match 4: Student vs MM_Improved Result: 4 to 16
Match 5: Student vs AB_Null     Result: 10 to 10
Match 6: Student vs AB_Open     Result: 7 to 13
Match 7: Student vs AB_Improved Result: 5 to 15
```

Results:

```
-----
Student          37.86%
```

3 Discussion

The only method to beat ID_Improved was the number of available moves. The only difference this had to the ID_improved is the priority coefficient. While this method is simple it is fast allowing for many layers of depth to be explored in minimax/iterative deepening.

Initially the area based heuristic was thought to be promising but its dismal performance is likely due to two main factors, it is computationally expensive and only captures valuable information later in the game. Initially there will only be a single cluster of empty space and any move within this heuristic will not reflect whether a move is good or bad except at points where sections of the board can be cut off.

The axis based heuristic also performs poorly, and this surprising as it is the algorithm that is most likely to generate an initially centered move which is one of the more strategically sound opening moves. The disparity in score to the movement heuristic are surprising given their similarities. Given that this is more computationally expensive there was always a chance that it was going to perform slightly worse however the difference in score (37.86%*vs*78.57%) is far too great for that. The most likely explanation is that the fundamental assumption on which it is based – maximizing movement in all axis is flawed. This makes sense as this would cause the agent to try to stay as central to the board as possible, not strategically sound. Perhaps if the author had been less sleep deprived when conceiving of this he would have realized before implementation...

4 Conclusion

The best heuristic to use is the number of moves available to the agent taking less the scaled number of opponent moves. It abstractly captures the fundamental concept of the game, from a high level allowing minimax/iterative deepening to do the heavy bulk of the work. It is likely not the best strategy, different stages of the game may require different strategies. In fact an obvious improvement might be to make the opponent priority coefficient scale with game turn, as this may become more or less relevant as the game progresses.