Heuristic Analysis

Three custom score functions are considered in this analysis. In the so-called improved scoring the opponent's number of available moves, shown as m_o here, is subtracted from the player's available moves, shown as m_p:

$$score = m_p - 2\, m_o$$

All score functions considered in this analysis are slightly modified version of this improved score, which will be referred to as IS11. The first two are linear functions of m_p and m_o, while the third one is non-linear but monotonous. Below the score functions and the tournament results are briefly introduced.

custom_score

This custom score is like improved score (IS11) except that the negative term or m_o is multiplied by a factor of 2. We show it with IS12 to simply represent multiply factor of 1 and 2 for m_p and m_o.

$$score = m_p - 2\, m_o$$

custom_score_2

Similar to improved scores IS11 and IS12, but here player score factor is 2.

$$score = 2\, m_p - m_o$$

custom_score_3

One other approach for scoring used here is to normalize the score defined in improved score, IS11, by the sum of the available move: $m_p + m_o$.

$$score = (m_p - m_o)/(m_p + m_o)$$

In some way this is similar to IS12 as it can distinguish between two cases where the difference between m_p and m_o is the same but different values of m_o will result in different score. For example consider the case 1 in which $m_p = 8$ and $m_o = 7$, and case 2 in which $m_p = 4$ and $m_o = 3$.

Improved score for both case 1 and case 2 is $score_{IS11} = m_p - m_o = 1$. The function custom_score_1 assigns different values to these two cases:

$$\text{Case 1: } score = m_p - 2\, m_o = 8 - 14 = -6$$

$$\text{Case 2: } score = m_p - 2\, m_o = 4 - 6 = -2$$

The function custom_score_2 also assigns different values to these two cases:

$$\text{Case 1: } score = (m_p - m_o)/(m_p + m_o) = (8 - 7)/(8 + 7) = 1/15$$

$$\text{Case 2: } score = (m_p - m_o)/(m_p + m_o) = (4 - 3)/(4 + 3) = 1/7$$

As we see the scores assigned to these cases are different both in custom_score_1 and custom_score_2. This can be important because one move difference weighs more at the end of the game when most spaces on the board are taken and there aren't many moves left and this is properly presented with these functions.

Results

The tournament table of results shows all these three score functions have slightly better performance than simple improved score, but the difference between the three methods and even the apparent advantage to the simple improved score is rather small and it could partly be due to the randomness in the games and the tournament! They all perform superbly against minimax players and when the opponent uses alphabeta pruning, they do win most of the time but at a much lower rate.

A better performance may be achieved if a more complicated evaluation function is used depending on the stage of the game, i.e. beginning, middle and end. Among the three evaluation functions presented here, I would recommend the third one for following reasons:

- Although the score function is non-linear, it is still monotonous and two added operations needed to calculate the score is negligible.
- The score function is normalized and as a result the scores are limited between -1 and 1. This allows in a more sophisticated programming comparison of the scores among different depths.
- The overall performance against more challenging opponents, i.e. AB player, is better than other score functions.

****
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	10	0
2	MM_Open	5	5	9	1	9	1	8	2
3	MM_Center	10	0	8	2	10	0	8	2
4	MM_Improved	5	5	8	2	7	3	7	3
5	AB_Open	5	5	5	5	8	2	7	3
6	AB_Center	5	5	5	5	3	7	7	3
7	AB_Improved	7	3	5	5	5	5	4	6
	Win Rate:	65.7%		71.4%		72.9%		72.9%	